ยูนิกซ์เบื้องต้น

Introduction to Unix

โดย

เครือข่ายซอฟต์แวร์โอเพนซอร์ส

Open Source Software Network

2550

คำนำ

หนังสือยูนิกซ์เบื้องต้นเล่มนี้ มีวัตถุประสงค์ให้ นักเรียน นิสิตนักศึกษา และผู้ที่สนใจใช้เพื่อศึกษาหาความรู้เกี่ยวกับการใช้งานระบบปฏิบัติการยูนิกซ์เบื้องต้น

เนื้อหาของหนังสือเล่มนี้ แบ่งเป็น 6 บท ประกอบด้วย ประวัติความเป็นมาของยูนิกซ์ โครงสร้างของไฟล์บนระบบปฏิบัติการยูนิกซ์ คำสั่งที่ควรรู้ สิทธิการเข้าถึงระบบแฟ้มข้อมูล (Permission) Shell เบื้องต้น และการใช้งานโปรแกรม vi เบื้องต้น ซึ่งได้รวบรวมและเรียบเรียงสาระสำคัญที่เกี่ยวข้องให้ครอบคลุม และเป็นประโยชน์มากที่สุด รวมถึงภาคผนวกของหนังสือเล่มนี้ จะเป็นส่วนที่ช่วยให้เข้าถึงระบบปฏิบัติการยูนิกซ์ ทั้งนี้ขึ้นอยู่กับสภาพแวดล้อมของผู้ใช้ ในกรณีที่ผู้ใช้ติดต่อเข้าใช้เครื่องแม่ข่ายยูนิกซ์ผ่านเครื่องลูกข่าย (Client) ที่เป็นไมโครซอฟต์วินโดว์ สามารถศึกษาวิธีการเข้าถึงเครื่องแม่ข่าย (Server) ด้วยโปรแกรม WinSCP จากภาคผนวก ก และโปรแกรม PuTTY จากภาคผนวก ข สำหรับภาคผนวก ค เป็นตัวอย่างวิธีการตั้งค่า BIOS เพื่อบูตใช้งานยูนิกซ์จาก LiveCD ส่วนภาคผนวก ง เป็นวิธีการตั้งค่า BIOS ที่ต่างชนิดกันเพื่อใช้งานยูนิกซ์จาก Flash Drive อย่างไรก็ตามการมีเครื่องแม่ข่าย (Server) ยูนิกซ์ไว้ใช้งานจะทำให้สามารถใช้ประโยชน์จากซอฟต์แวร์โอเพนซอร์สได้มากกว่า

หวังเป็นอย่างยิ่งว่าหนังสือเล่มนี้คงมีประโยชน์แก่ทุกท่าน หากมีข้อเสนอแนะและข้อผิดพลาดประการใด คณะผู้จัดทำยินดีน้อมรับจักขอบพระคุณยิ่ง

เครือข่ายซอฟต์แวร์โอเพนซอร์ส

กรกฎาคม 2550

สารบัญ

บทที่ 1 ประวัติความเป็นมาของยูนิกซ์

- ประวัติความเป็นมาของยูนิกซ์..... 1
- ข้อดีของระบบปฏิบัติการยูนิกซ์..... 1

บทที่ 2 โครงสร้างของไฟล์บนระบบปฏิบัติการยูนิกซ์

- โครงสร้างไดเรคทอรีของระบบปฏิบัติการยูนิกซ์โดยทั่วไป..... 2

บทที่ 3 คำสั่งที่ควรรู้จัก

- การ Login และ Logout..... 5
- คำสั่งในการจัดการไฟล์..... 6
- คำสั่งในการจัดการไดเรกทอรี่..... 13
- คำสั่งในการจัดการเกี่ยวกับผู้ใช้..... 17
- คำสั่งในการจัดการติดต่อสื่อสาร..... 20
- คำสั่งในการค้นหาไฟล์ และระบบรักษาความปลอดภัย..... 21
- คำสั่งพื้นฐานอื่น ๆ..... 24

บทที่ 4 สิทธิการเข้าถึงของระบบแฟ้มข้อมูล

- สิทธิการเข้าถึงของระบบแฟ้มข้อมูล..... 26

บทที่ 5 Shell เบื้องต้น

- โครงสร้างพื้นฐานการทำงานของระบบปฏิบัติการยูนิกซ์..... 29
- ประเภทของเชลล์ที่นิยมใช้ในปัจจุบัน..... 30
- การเปลี่ยนทิศทางของข้อมูลเข้าและผลลัพธ์ที่ได้จากการรันShell Script..... 34

สารบัญ (ต่อ)

บทที่ 6 การใช้งานโปรแกรม vi เบื้องต้น

- หน้อจอโปรแกรม vi..... 36
- เริ่มต้นการใช้งานโปรแกรม vi..... 37
- คำสั่งในการเลื่อนเคอเซอร์และหน้าจอขณะอยู่ในโหมดคำสั่ง..... 37
- คำสั่งในการเข้าสู่โหมดพิมพ์ข้อความ..... 39
- คำสั่งพิมพ์ข้อความทับหรือเปลี่ยนแทนข้อความ (Replace)..... 40
- การยกเลิกคำสั่ง (Undo)..... 40
- การลบตัวอักษร..... 40
- การลบตัวบรรทัด..... 40
- การลบข้อความ..... 41
- การคัดลอกและวาง..... 41
- การค้นหาคำ..... 41
- การบันทึกข้อมูลลงไฟล์..... 42

ภาคผนวก ก การใช้โปรแกรมประยุกต์เพื่อการถ่ายโอนแฟ้มข้อมูล..... 43

ภาคผนวก ข การใช้โปรแกรมประยุกต์เพื่อการเข้าใช้งาน Text Shell..... 53

ภาคผนวก ค การใช้งานจากแผ่น LiveCD..... 57

ภาคผนวก ง การใช้งานจากอุปกรณ์สำหรับบันทึกข้อมูลชนิด Flash Drive.....62

บทที่ 1

ประวัติความเป็นมาของยูนิกซ์

- ในปี 1970 ระบบปฏิบัติการยูนิกซ์ได้เกิดขึ้น เป็นระบบที่รับความนิยมมาก มีการใช้งานกันอย่างแพร่หลาย ไม่ว่าจะเป็นสถานศึกษา หรือสถานที่ทำงาน
- ในปี 1973 ได้เขียนยูนิกซ์ขึ้นมาใหม่ ด้วยภาษาซีทำให้สะดวกต่อการนำยูนิกซ์ไปใช้งานบนเครื่องชนิดอื่น ยูนิกซ์ได้ถูกนำไปพัฒนาต่อ และมีระบบที่ได้รับความนิยม คือ BSD และ System V
- ในปัจจุบันมีระบบปฏิบัติการแบบยูนิกซ์เกิดขึ้นมากมาย เช่น Linux FreeBSD NetBSD ฯลฯ และระบบปฏิบัติการยูนิกซ์ก็ยังได้รับความนิยมกันอย่างแพร่หลาย
- มีการพัฒนาระบบปฏิบัติการลีนุกซ์ของประเทศไทยหลายชนิด เช่น บูรพาลีนุกซ์ ลีนุกซ์ทะเล ลีนุกซ์สิส เป็นต้น

ของดีของระบบปฏิบัติการยูนิกซ์

- ระบบปฏิบัติการยูนิกซ์ มีความยืดหยุ่นในการใช้งานสูง สามารถใช้กับงานเล็กๆ ไปจนถึง งานที่มีขนาดใหญ่
- สามารถทำงานได้พร้อมกันทีละหลายๆคน
- สามารถทำงานที่ซับซ้อนได้ดีกว่าระบบปฏิบัติการอื่นๆ
- ทำงานหลายๆ อย่างพร้อมกันได้ในเวลาเดียวกัน ทำให้ไม่ต้องเสียเวลาในการรอ โดยแบ่งการทำงานออกเป็น Foreground และ Background
- สามารถรันได้บนแพลตฟอร์มหลายระดับ
- เป็นซอฟต์แวร์โอเพนซอร์ส

บทที่ 2

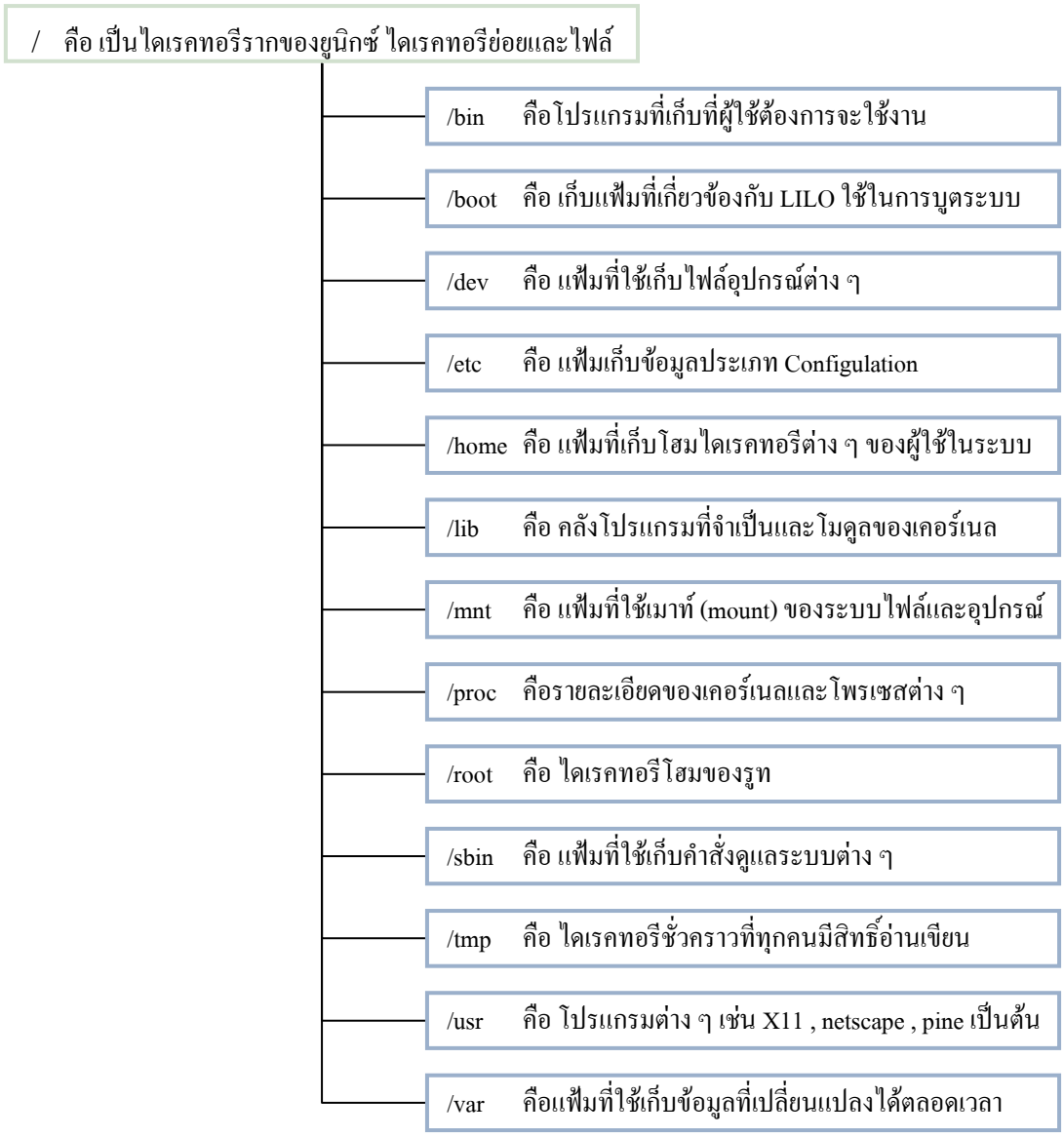
โครงสร้างของไฟล์บนระบบปฏิบัติการยูนิกซ์

โครงสร้างไดเรคทอรีของระบบปฏิบัติการยูนิกซ์โดยทั่วไป

ในช่วงกลางปี คศ.1993 มีการออกแบบโครงสร้างและไดเรคทอรีของยูนิกซ์เป็นแบบ FSSTND (File System Standard) ต่อมาได้มีการพัฒนาโครงสร้างใหม่เป็นแบบ Filesystem Hierarchy Standard : FHS ซึ่งมีโครงสร้างคล้ายกับต้นไม้ (Tree) และมีการจัดแบบลำดับชั้น จะแสดงดังรูปที่ 2-1

โปรแกรม และข้อมูล ของระบบปฏิบัติการ ที่ผู้ใช้ทุกคนต้องพบ ระบบแฟ้มประกอบด้วย 2 ส่วน คือ Collection of files ซึ่งเก็บข้อมูลที่สัมพันธ์กัน และไดเรคทอรีstructure สำหรับจัดการ และให้ข้อมูลแฟ้มทั้งหมดในระบบ บางระบบปฏิบัติการมีส่วนที่ 3 คือ Partitions ซึ่งแยก Physically หรือ Logically ของระบบไดเรคทอรีโดยเนื้อหาในบทนี้จะกล่าวถึงแฟ้ม และโครงสร้างไดเรกทรอรี รวมถึงการป้องกันแฟ้ม จากการเข้าถึงในระบบ Multiple users และระบบ File sharing

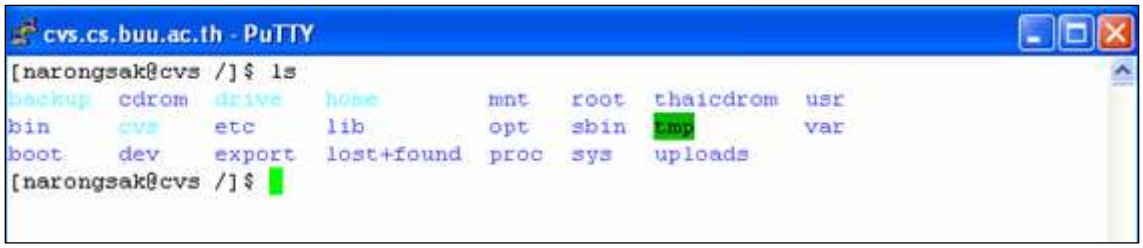
วิธีการจัดเก็บข้อมูลที่ใช้กันในระบบปฏิบัติการทุกชนิดคือ จัดเก็บข้อมูลเป็นแฟ้มข้อมูลหรือไฟล์ (file) ไฟล์คือสิ่งที่บรรจุข้อมูล, โปรแกรมหรืออะไรก็ได้ที่ผู้ใช้ต้องการรวบรวมไว้เป็นชุดเดียวกัน การอ้างถึงไฟล์หรือข้อมูลต่าง ๆ ภายในไฟล์ของโปรแกรม จะไม่มีความเกี่ยวข้องกับแอดเดรสของโปรแกรมใด ๆ ทั้งสิ้น ระบบปฏิบัติการมีโอเปอร์เรชั่นพิเศษที่เรียกว่า system call ไว้ให้โปรแกรมเรียกใช้ เพื่อให้สามารถจัดการงานที่เกี่ยวกับไฟล์ได้


/ คือ เป็นไดเรคทอรีรากของยูนิกซ์ ไดเรคทอรีย่อยและไฟล์
/bin คือโปรแกรมที่เก็บที่ผู้ใช้ต้องการจะใช้งาน
/boot คือ เก็บแฟ้มที่เกี่ยวข้องกับ LILO ใช้ในการบูตระบบ
/dev คือ แฟ้มที่ใช้เก็บไฟล์อุปกรณ์ต่าง ๆ
/etc คือ แฟ้มเก็บข้อมูลประเภท Configulation
/home คือ แฟ้มที่เก็บโฮมไดเรคทอรีต่าง ๆ ของผู้ใช้ในระบบ
/lib คือ คลังโปรแกรมที่จำเป็นและโมดูลของเคอร์เนล
/mnt คือ แฟ้มที่ใช้เมาท์ (mount) ของระบบไฟล์และอุปกรณ์
/proc คือรายละเอียดของเคอร์เนลและโพรเซสต่าง ๆ
/root คือ ไดเรคทอรีโฮมของรูท
/sbin คือ แฟ้มที่ใช้เก็บคำสั่งดูแลระบบต่าง ๆ
/tmp คือ ไดเรคทอรีชั่วคราวที่ทุกคนมีสิทธิ์อ่านเขียน
/usr คือ โปรแกรมต่าง ๆ เช่น X11 , netscape , pine เป็นต้น
/var คือแฟ้มที่ใช้เก็บข้อมูลที่เปลี่ยนแปลงได้ตลอดเวลา

รูปที่ 2-1 โครงสร้างไดเรคทอรีของระบบปฏิบัติการยูนิกซ์แบบมาตรฐาน File system Hierarchy Standard (FHS)

โครงสร้างไฟล์นี้เป็นโครงสร้างระบบไฟล์ของ ยูนิกซ์และลีนุกซ์โดยทั่วไป ซึ่งบางครั้งอาจจะมีมากหรือน้อยกว่านี้ จะขึ้นอยู่กับผู้ดูแลระบบเป็นผู้จัดการ เช่น บางเครื่องแม่ข่าย (Server) อาจจะมีการแบ่ง home ออกเป็น home1 และ home2 เพื่อจัดการกับผู้ใช้ในแต่ละประเภทเป็นต้น

ต่อไปจะเป็นการอธิบายโครงสร้างของระบบแฟ้มข้อมูล โดยใช้คำสั่งแบบสั่งงานด้วยคำสั่ง (Command Line) ในการสาธิตโครงสร้างระบบแฟ้มข้อมูลของระบบปฏิบัติการลีนุกซ์ โดยเครื่องแม่ข่าย (Server) ที่จะใช้คือ cvs.buu.ac.th ซึ่งเป็นระบบปฏิบัติการบูรพาลีนุกซ์ (Burapha Linux Server) ซึ่งจะมีโครงสร้างเป็นมุมมองไดเรคทอรีดังรูปที่ 2-2



รูปที่ 2-2 แสดงโครงสร้างไดเรคทอรีของระบบปฏิบัติการบูรพาลีนุกซ์ (Burapha Linux Server)

โดยผู้ใช้ที่ทำการเข้าใช้งานนั้นชื่อ narongsak และเข้าใช้งานที่เครื่อง CVS สัญลักษณ์ที่ใช้นี้จะเรียกว่า พรอมท์ (Prompt) สังเกตได้จากบรรทัดแรกของรูปที่ 2-2 เครื่องหมาย / แสดงตำแหน่งในระบบที่ผู้ใช้กำลังดำเนินการอยู่ ในที่นี้จะเรียกตำแหน่ง / นี้ว่า รูทของระบบ (root) และ Prompt นี้จะจบด้วยเครื่องหมายดอลล่า $

ต่อจากเครื่องหมาย $ คือคำสั่งของระบบปฏิบัติการยูนิกซ์ (ซึ่งจะกล่าวในรายละเอียดต่อไป) คำสั่งที่พิมพ์ลงไปในส่วนของพรอมท์ คือ ls โดยคำสั่งนี้จะใช้เพื่อแสดงข้อมูลว่า ณ ตำแหน่งนั้นประกอบด้วยข้อมูลไดเรคทอรีหรือไฟล์ชื่ออะไรบ้าง

การเข้าใช้งานระบบปฏิบัติการยูนิกซ์ สามารถเข้าใช้งานได้ 2 วิธี คือ วิธีแรกเข้าจากหน้าจอของเครื่องยูนิกซ์โดยตรง และใช้โปรแกรมประเภท Emulator ที่จะจำลองหน้าขอของเครื่องแม่ข่าย โดยในที่นี้ขอแนะนำการใช้งานโปรแกรม PuTTY (รายละเอียดการใช้โปรแกรม PuTTY อยู่ในภาคผนวก ข) ซึ่งมีขั้นตอนดังนี้

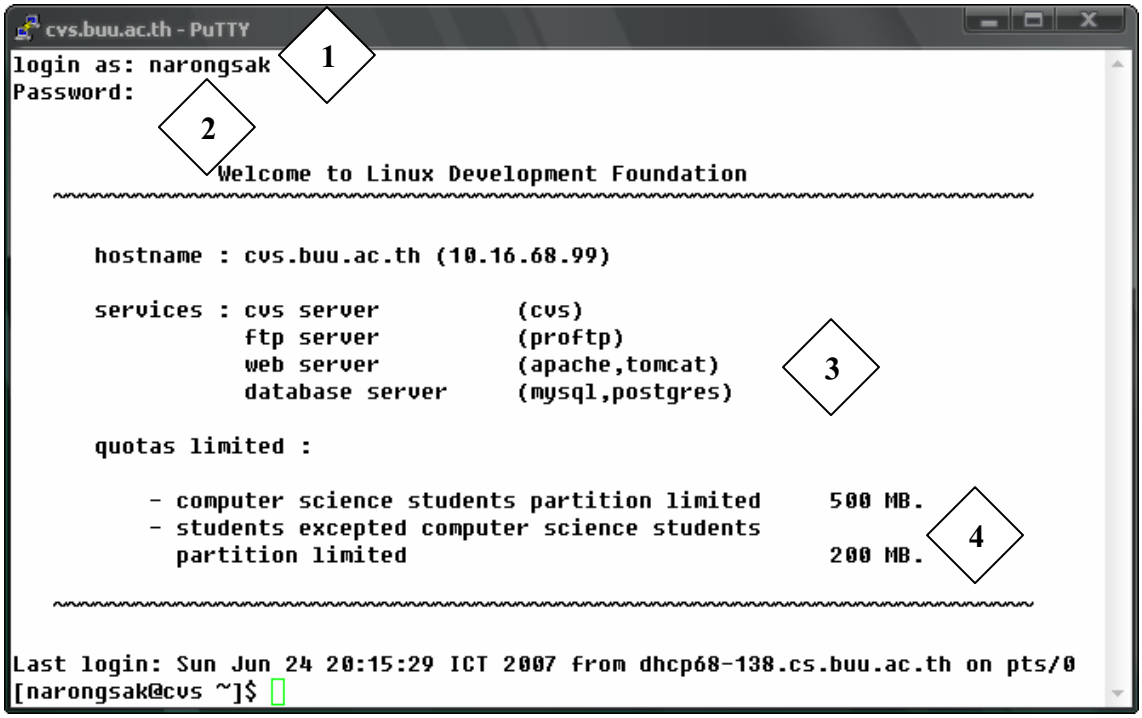
รูปที่ 2-3 ไอคอนโปแกรม PuTTY

บทที่ 3

คำสั่งที่ควรรู้

การ Login และ Logout

การ Login เป็นขั้นตอนแรกที่จะต้องทำก่อนที่จะเข้าใช้งานระบบปฏิบัติการยูนิกซ์ได้ โดยผู้ที่จะใช้งานจะต้องใส่ username และ password ก่อนการเข้าใช้เสมอ



รูปที่ 3-1 รูปแสดงการเข้าสู่ระบบ

บรรทัดแรกที่แสดงในรูปที่ 3-1 แสดงการเข้าใช้งานของผู้ใช้ชื่อ narongsak และบรรทัดที่ 2 จะเป็นการใส่รหัสผ่าน ซึ่งข้อพึงระวังคือ ในหน้าจอจะไม่มีการแสดงข้อความใดๆ ขณะที่พิมพ์ ดังนั้น ผู้ใช้ควรจะต้องตั้งใจพิมพ์รหัสผ่านให้ถูกต้อง ถ้าพิมพ์ผิดเกิน 3 ครั้ง โปรแกรมจะปิดโดยอัตโนมัติ ซึ่งผู้ใช้ก็จะต้องเรียกโปรแกรมขึ้นมาใหม่อีกครั้งหนึ่ง

ส่วนที่ 3 และ 4 แสดงข้อมูลรายละเอียดของระบบ รวมไปถึงจำนวนโควต้าที่เหลืออยู่สำหรับผู้ใช้ ข้อมูลใน 2 ส่วนนี้ อาจจะมีหรือไม่มีก็ได้ ขึ้นอยู่กับว่าผู้ดูแลระบบเครือข่ายนั้น ตั้งค่าให้มีการแสดงผลบอกหรือไม่

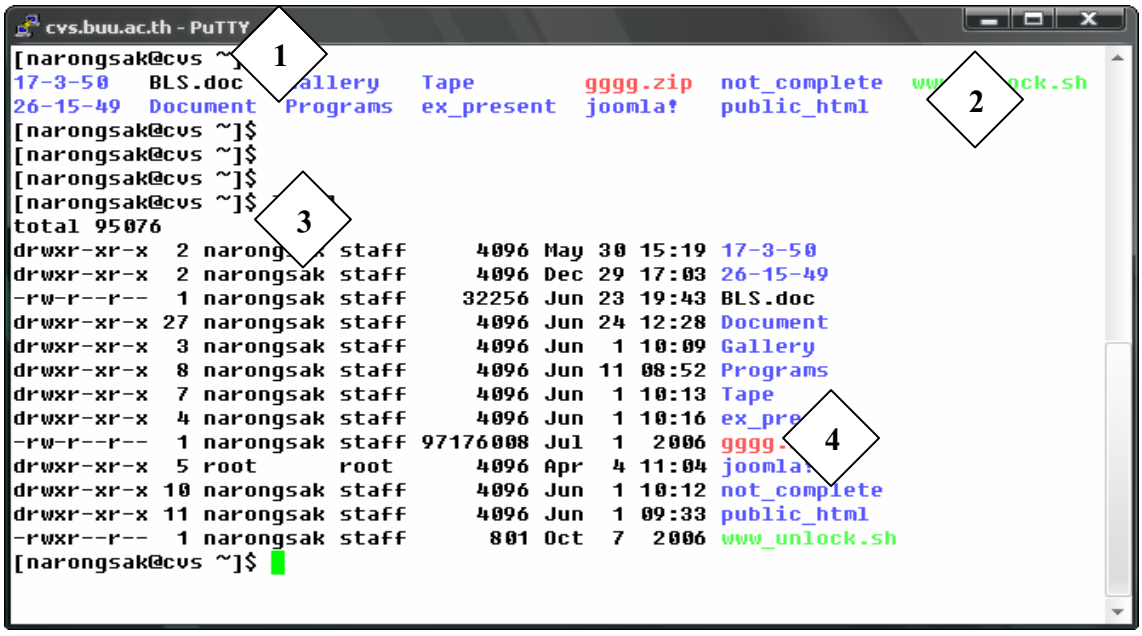
คำสั่งในการจัดการไฟล์

	คำสั่งในการจัดการไฟล์ : คำสั่ง ls
Syntax	**ls [OPTION] FILES**

เป็นคำสั่งที่ใช้สำหรับแสดงแฟ้มข้อมูล เมื่อเราใช้คำสั่ง ls โดยไม่มีเงื่อนไข จะเป็นการแสดงเฉพาะชื่อของไฟล์ต่างๆ ในไดเรคทอรีปัจจุบัน option ที่มักใช้กันใน ls คือ

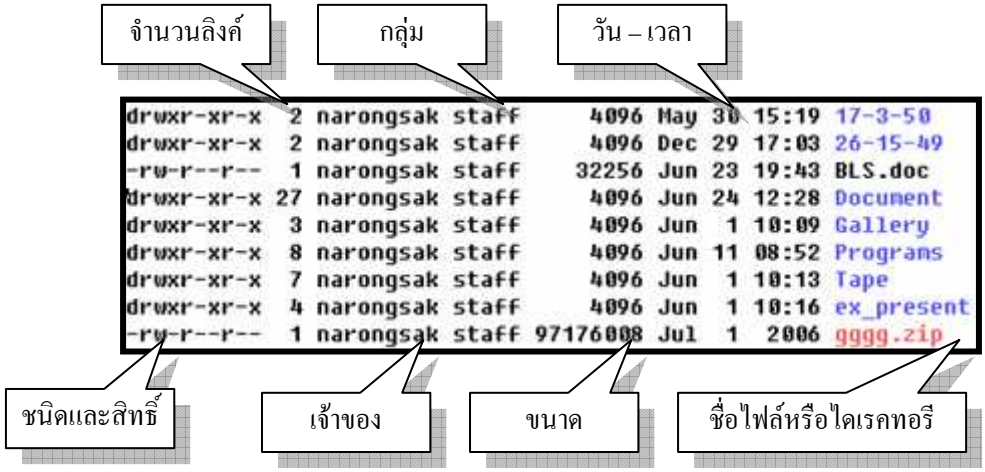
-l จะแสดงผลลัพธ์แบบเต็มรูปแบบ (Long Format)

-a จะแสดงแฟ้มข้อมูลทั้งหมด



รูปที่ 3-1 หน้าจอแสดงการใช้งานคำสั่ง ls –a และ ls –l

จากรูปที่ 3-2 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง ls จะแสดงแฟ้มและไดเรคทอรีในไดเรคทอรีปัจจุบันดังตำแหน่งที่ 2 ส่วนตำแหน่งที่ 3 เมื่อพิมพ์คำสั่ง ls -l จะแสดงข้อมูลเกี่ยวกับแฟ้ม และไดเรคทอรีในไดเรคทอรีปัจจุบัน โดยแสดงหนึ่งบรรทัดต่อหนึ่งไฟล์หรือไดเรคทอรีแบบเต็มรูปแบบ (long format) โดยแบ่งรายละเอียดออกเป็น 7 คอลัมภ์ ดังตำแหน่งที่ 4 โดยจะอธิบายดังรูปที่ 3-3



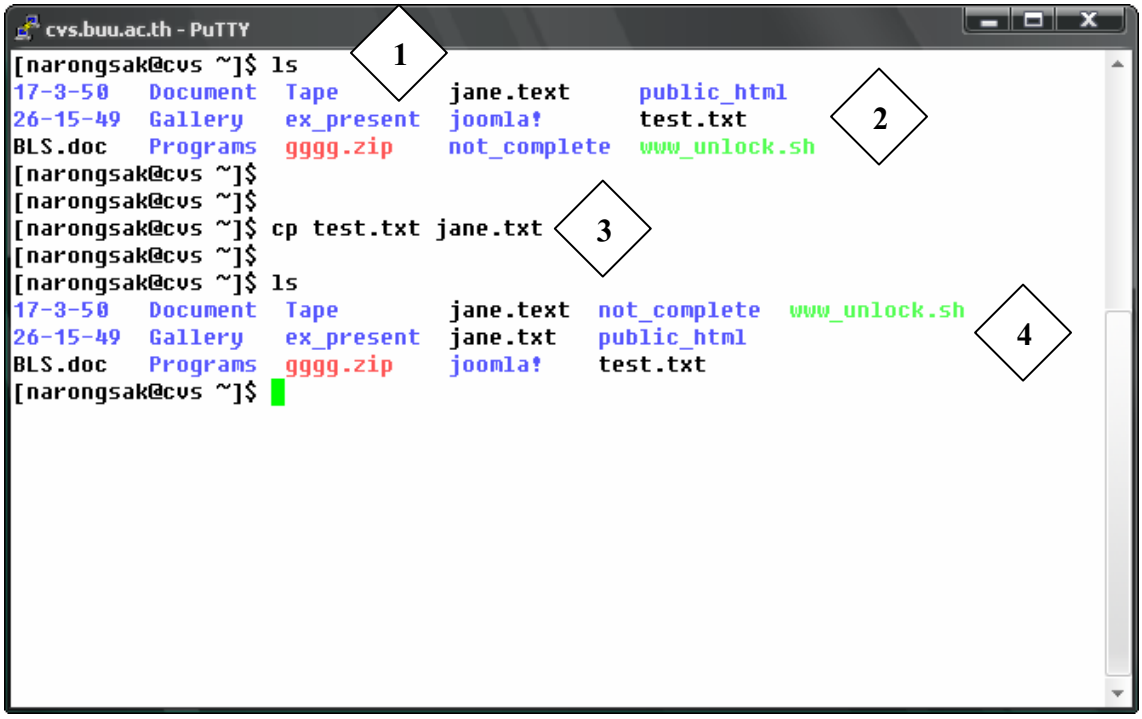
รูปที่ 3-3 แสดงรายละเอียดเมื่อใช้คำสั่ง ls -l

จากรูปที่ 3-3 โดยแต่ละคอลัมภ์มีความหมายดังนี้

- ชนิดและสิทธิ์ จะบ่งบอกประเภทของไฟล์นั้น ๆ ว่าเป็นชนิดใด โดยจะกำหนดเป็นอักขระ 10 ตัว ซึ่งอักขระตัวแรกบอกถึงชนิดของแฟ้มหรือไดเรคทอรี เช่น d ในบรรทัดแรกแสดงของไฟล์ชื่อ 17-3-50 เป็นไดเรคทอรี สำหรับ 9 อักขระหลังเป็นการกำหนดการอ่าน การเขียน และการใช้ไฟล์ ซึ่งกำหนดได้โดยใช้คำสั่ง chmod
- จำนวนลิงค์ จะบอกให้ทราบถึงจำนวนไฟล์ที่ลิงค์เข้ากับไดเรคทอรีหรือไฟล์นี้ ถ้าเป็นไดเรคทอรีว่างจะมีค่าระบุไว้เป็น 2
- เจ้าของ จะบอกให้ทราบว่าใครเป็นเจ้าของไฟล์หรือไดเรคทอรีนี้ เช่น narongsak
- ชื่อกลุ่ม บอกให้ทราบว่ารหัสผู้ใช้ผู้นี้มีสถานะอยู่ในกลุ่มใด
- ขนาด บอกให้ทราบถึงขนาดของไฟล์ว่ามีความจุเท่าไหร่ โดยแสดงเป็นไบต์ (Byte) แต่ถ้าเป็นไดเรคทอรีจะมีขนาดเท่ากับ 4096 ไบต์ (Byte) เสมอ
- วัน-เวลา จะแสดงวันและเวลาในการแก้ไขไฟล์หรือไดเรคทอรีล่าสุด
- ชื่อไฟล์ แสดงชื่อไฟล์ หรือไดเรคทอรี

	คำสั่งในการจัดการไฟล์ : คำสั่ง cp
Syntax	**cp [OPTION] [-T] SOURCE DEST** **cp [OPTION] SOURCE... DIRECTORY** **cp [OPTION] -t ไดเรคทอรีSOURCE.**
Example	**cp test.txt jane.txt**

เป็นคำสั่งที่ใช้สำหรับการคัดลอกไฟล์ รูปแบบการใช้คำสั่ง cp ชื่อของไฟล์ต้นทาง ตามด้วยชื่อของไฟล์ (ใหม่) ปลายทาง

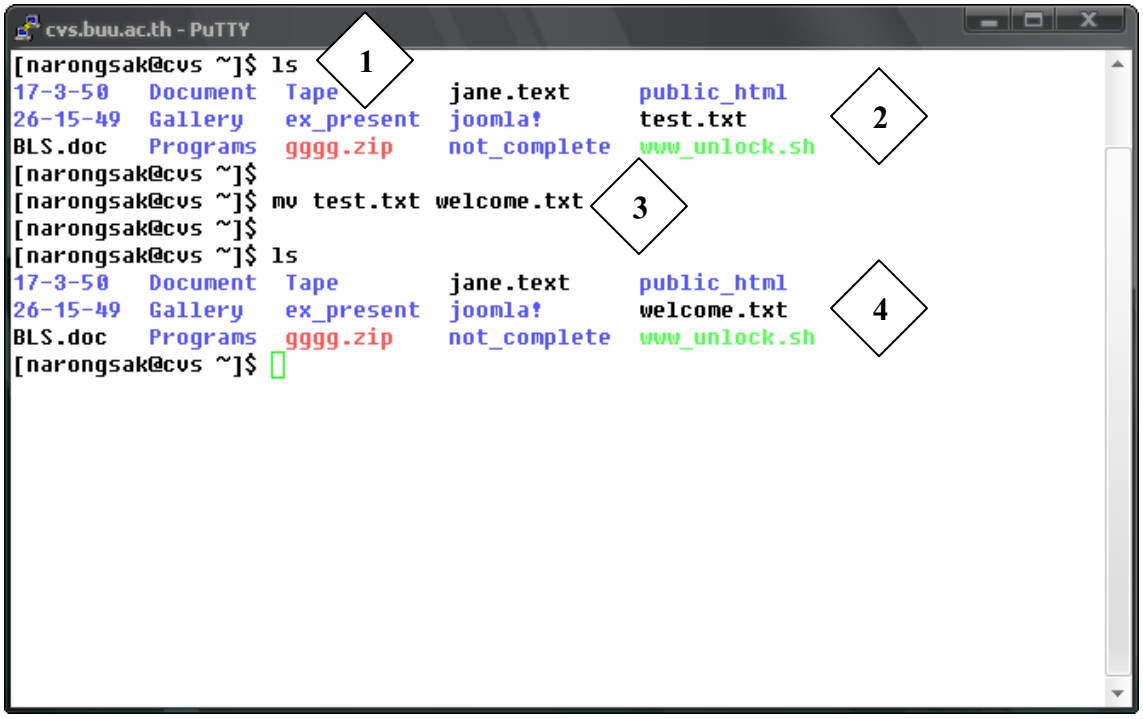


รูปที่ 3-4 หน้าจอแสดงการใช้งานคำสั่ง cp

จากรูปที่ 3.4 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง ls จะแสดงไฟล์และไดเรคทอรีทั้งหมดในไดเรคทอรีปัจจุบันดังตำแหน่งที่ 2 ทั้งนี้ต้องการคัดลอกไฟล์ test.txt เป็น ไฟล์ jane.txt สามารถทำได้โดยใช้คำสั่ง cp ดังตำแหน่งที่ 3 ซึ่งเมื่อใช้คำสั่ง ls อีกครั้งจะปรากฏไฟล์ jane.txt เพิ่มขึ้นมา

	คำสั่งในการจัดการไฟล์ : คำสั่ง mv
Syntax	**mv [OPTION]... [-T] SOURCE DEST** **mv [OPTION]... SOURCE... DIRECTORY** **mv [OPTION]... -tไดเรคทอรีSOURCE...**
Example	**mv test.txt welcome.txt**

คำสั่ง mv (move) ใช้สำหรับการย้ายแฟ้มข้อมูลและไดเรคทอรีรวมถึงการเปลี่ยนชื่อด้วย ทำงานเช่นเดียวกับคำสั่ง renameในดอส

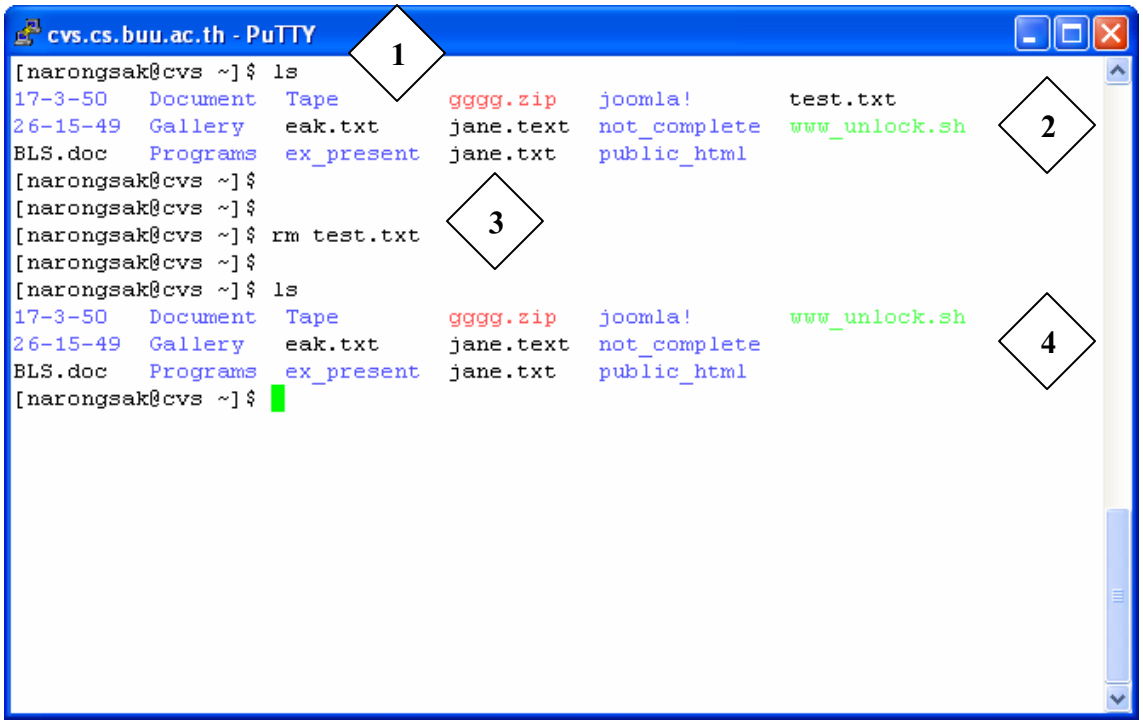


รูปที่ 3-5 หน้าจอแสดงการใช้งานคำสั่ง mv

จากรูปที่ 3-5 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง ls โดยจะแสดงไฟล์และไดเรคทอรีทั้งหมดในไดเรคทอรีปัจจุบันดังตำแหน่งที่ 2 ทั้งนี้ต้องการเปลี่ยนชื่อไฟล์ text.txt เป็น welcome.txt สามารถทำได้โดยใช้คำสั่ง mv ดังตำแหน่งที่ 3 ซึ่งเมื่อใช้คำสั่ง ls อีกครั้งจะปรากฏว่า ไฟล์ text.txt ไม่ปรากฏ โดยจะพบแต่ไฟล์ welcome.txt

	คำสั่งในการจัดการไฟล์ : คำสั่ง rm
Syntax	**rm [OPTION] [FILE]**
Example	**rm test.txt**

คำสั่ง rm (romove) มีไว้ใช้สำหรับลบไฟล์ ทั้งแบบไฟล์เดียวหรือแบบกลุ่มของไฟล์ได้ โดยใช้คำสั่งเพียงครั้งเดียว

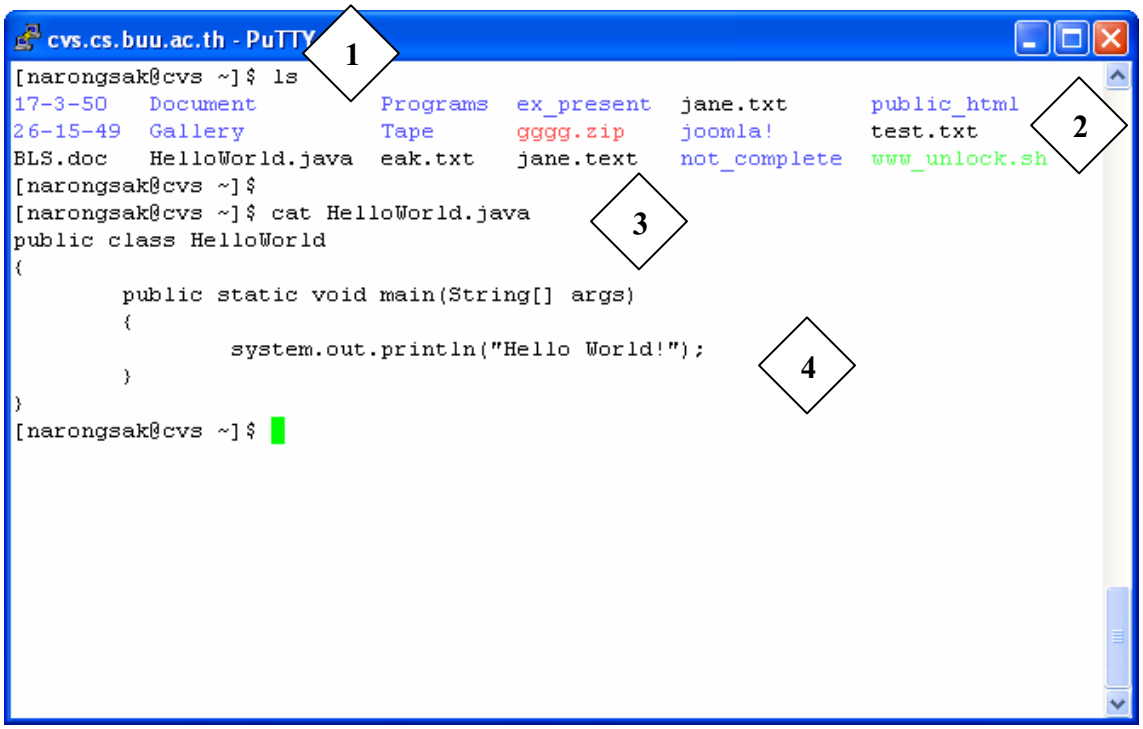


รูปที่ 3-6 หน้าจอแสดงการใช้งานคำสั่ง rm

จากรูปที่ 3-6 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง ls โดยจะแสดงไฟล์และไดเรคทอรีทั้งหมดในไดเรคทอรีปัจจุบันดังตำแหน่งที่ 2 ทั้งนี้ต้องการลบไฟล์ test.txt เป็น welcome.txt สามารถทำได้โดยใช้คำสั่ง rm ดังตำแหน่งที่ 3 ซึ่งเมื่อใช้คำสั่ง ls อีกครั้งจะปรากฏว่า ไฟล์ test.txt ได้ถูกลบไปแล้ว

	คำสั่งในการจัดการไฟล์ : คำสั่ง cat (concatenate)
Syntax	**cat [OPTION] [FILE]**
Example	**cat HelloWorld.java**

คำสั่ง cat (concatenate) มีหน้าที่แสดงผลรายระเอียดภายในของไฟล์ทางจอภาพ



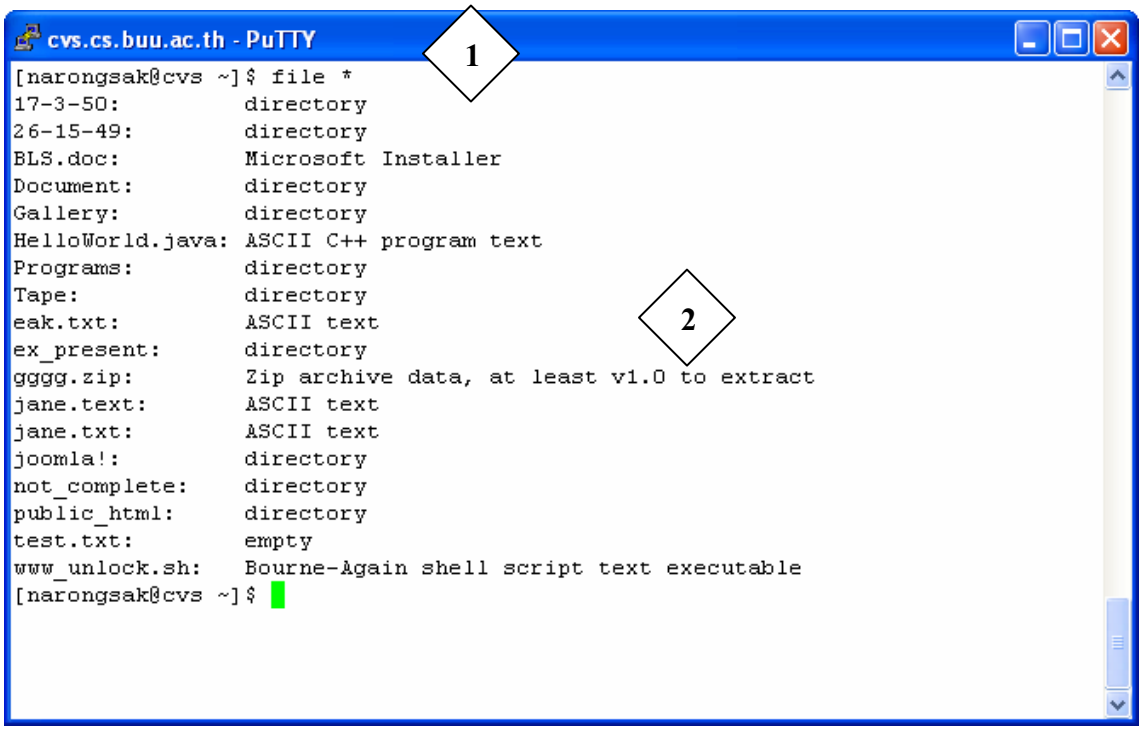
รูปที่ 3-7 หน้าจอแสดงการใช้งานคำสั่ง cat

จากรูปที่ 3-7 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง ls โดยจะแสดงไฟล์และไดเรคทอรีทั้งหมดในไดเรคทอรีปัจจุบันดังตำแหน่งที่ 2 ทั้งนี้ต้องการดูรายละเอียดของไฟล์ HelloWorld.java สามารถทำได้โดยใช้คำสั่ง cat ดังตำแหน่งที่ 3 ซึ่งรายละเอียดของไฟล์จะปรากฏออกทางหน้าจอ

	คำสั่งในการจัดการไฟล์ : คำสั่ง cat (concatenate)
Syntax	**file [OPTION] [FILE]**
Example	**file ***

คำสั่งfile จะทำการแสดงชยิดของไฟล์ที่มีใน ไดเรคทอรี ซึ่งคำสั่ง file จะทำการอ่าน Content และบอกประเภทของแฟ้มข้อมูลนั้นๆ

รูปแบบการใช้คำสั่ง file [option]... file



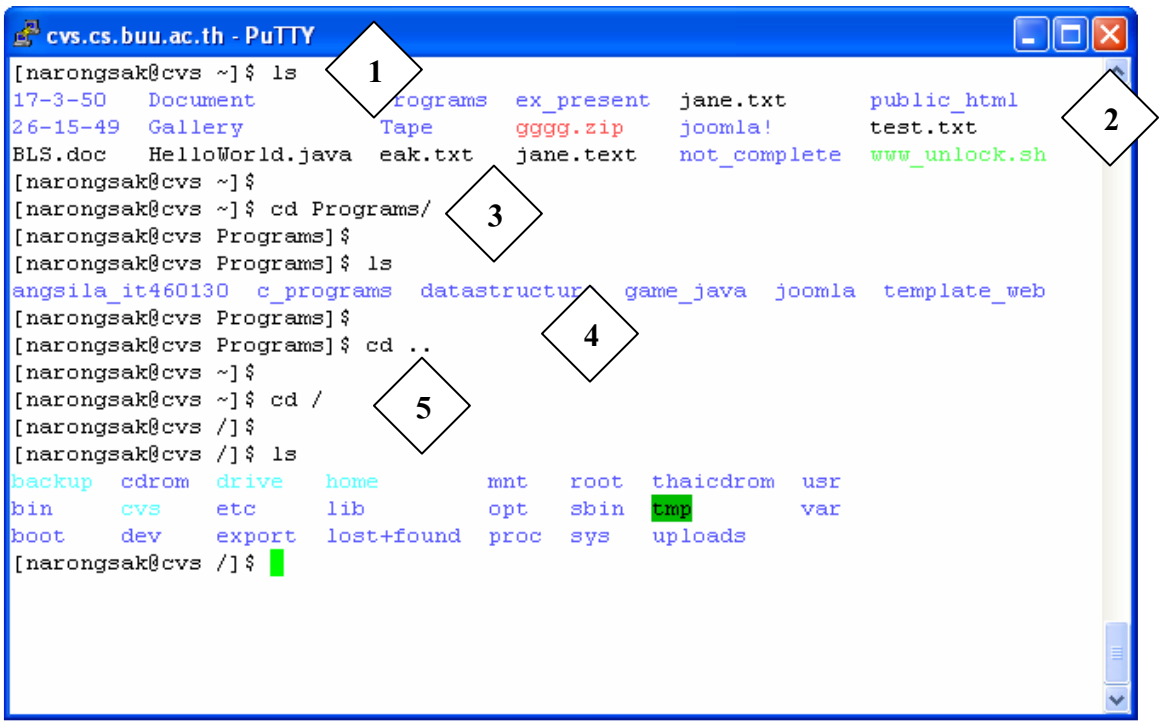
รูปที่ 3-8 หน้าจอแสดงการใช้งานคำสั่ง file

จากรูปที่ 3-8 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง file * โดยระบบจะแสดงไฟล์ทั้งหมดในไดเรคทอรีปัจจุบันพร้อมทั้งบอกประเภทของไฟล์ ดังตำแหน่งที่ ซึ่งรายละเอียดของไฟล์จะปรากฏออกทางหน้าจอ

คำสั่งในการจัดการไดเรคทอรี

	คำสั่งในการจัดการไดเรคทอรี : คำสั่ง cd
Syntax	**cd [DIRECTORY]**
Example 1	**cd Programs** คือ เข้าไปในไดเรคทอรี Programs
Example 2	**cd ..** คือ กลับไปที่ ไดเรคทอรีก่อนหน้า
Example 3	**cd /** คือ ไปที่ ไดเรคทอรี root

คำสั่ง cd (Change Directory) เป็นคำสั่งที่ใช้ในการเปลี่ยนไดเรคทอรี

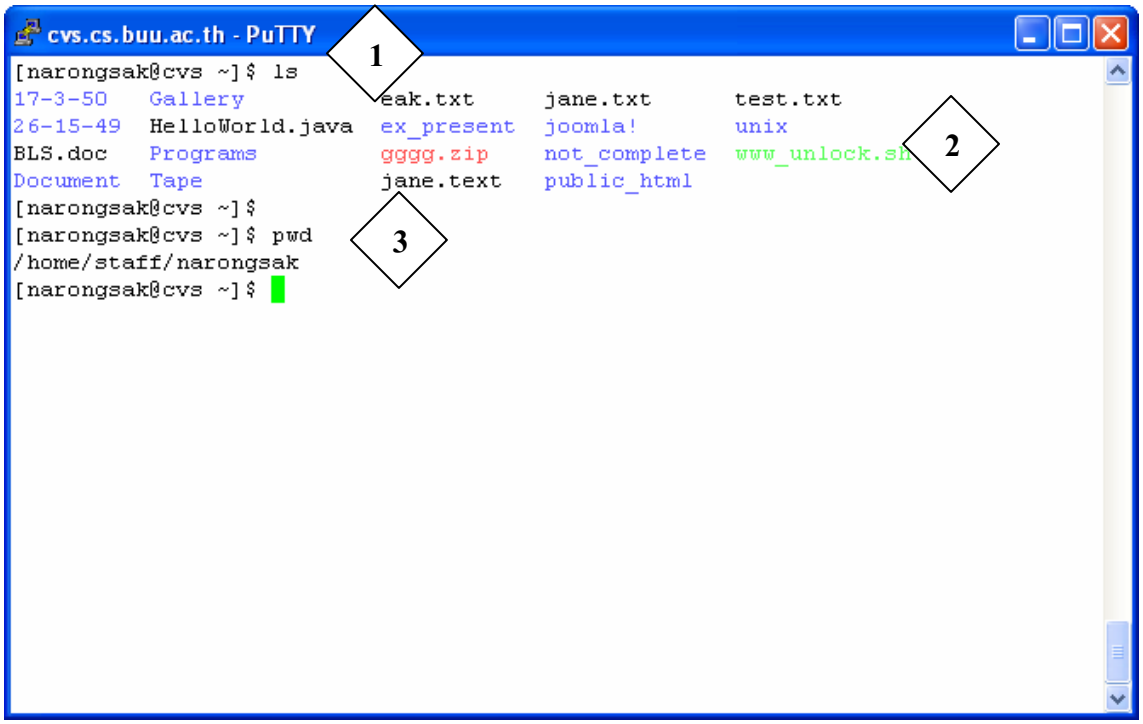


รูปที่ 3-9 หน้าจอแสดงการใช้งานคำสั่ง cd

จากรูปที่ 3-9 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง ls โดยจะแสดงไฟล์และไดเรคทอรีทั้งหมดในไดเรคทอรีปัจจุบันดังตำแหน่งที่ 2 ทั้งนี้ต้องการเข้าไดเรคทอรี Programs สามารถทำได้โดยใช้คำสั่ง cd Programs ดังตำแหน่งที่ 3 ซึ่งเมื่อใช้คำสั่ง ls อีกครั้งจะปรากฏว่า ไฟล์ต่างๆที่อยู่ในไดเรคทอรี Programs และถ้าต้องการออกจากไดเรคทอรี สามารถทำได้โดยใช้คำสั่ง cd .. ดังตำแหน่งที่ 4 และเมื่อต้องการออกไปยังไดเรคทอรี root สามารถทำได้โดยใช้คำสั่ง cd \ ดังตำแหน่งที่ 5 ซึ่งเมื่อใช้คำสั่ง ls อีกครั้งจะปรากฏไฟล์ต่างที่อยู่ในไดเรคทอรี root

	คำสั่งในการจัดการไดเรคทอรี : คำสั่ง pwd
Syntax	**pwd [OPTION]**
Example	**pwd**

เป็นคำสั่งที่ใช้สำหรับแสดงไดเรคทอรีปัจจุบัน (ในทำนองเดียวกับการพิมพ์ cd บน DOS) มาจากคำว่า print work ไดเรคทอรี



รูปที่ 3-10 หน้าจอแสดงการใช้งานคำสั่ง pwd

จากรูปที่ 3-10 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง ls โดยจะแสดงไฟล์และไดเรคทอรีทั้งหมดในไดเรคทอรีปัจจุบันดังตำแหน่งที่ 2 ทั้งนี้ต้องการที่จะรู้ตำแหน่งของไดเรคทอรี สามารถทำได้โดยใช้คำสั่ง pwd ระบบจะแสดงที่อยู่ของไดเรคทอรีปัจจุบัน

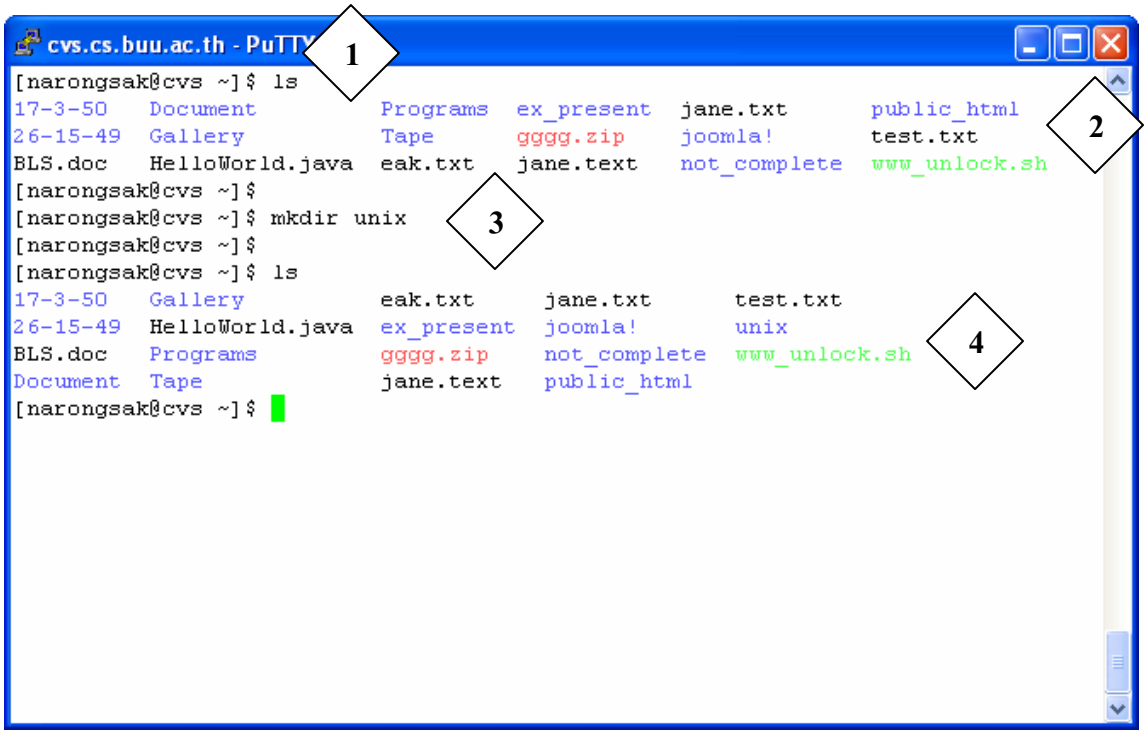
	คำสั่งในการจัดการไดเรคทอรี : คำสั่ง mkdir
Syntax	**mkdir [OPTION] [FILE]**
Example	**mkdir unix**

เป็นคำสั่งสำหรับใช้ในการสร้างไดเรคทอรี(makeไดเรคทอรี) เมื่อเราต้องการไดเรคทอรีใหม่ ให้ใช้คำสั่ง mkdir ตามด้วยชื่อไดเรคทอรีใหม่

โดย option ส่วนใหญ่ที่นิยมใช้กัน คือ

-m จะทำการกำหนด Permissioin

-p จะทำการสร้าง Parentไดเรคทอรีให้ด้วยกรณีที่ยังไม่มีการระบุไดเรคทอรี



รูปที่ 3-11 หน้าจอแสดงการใช้งานคำสั่ง mkdir

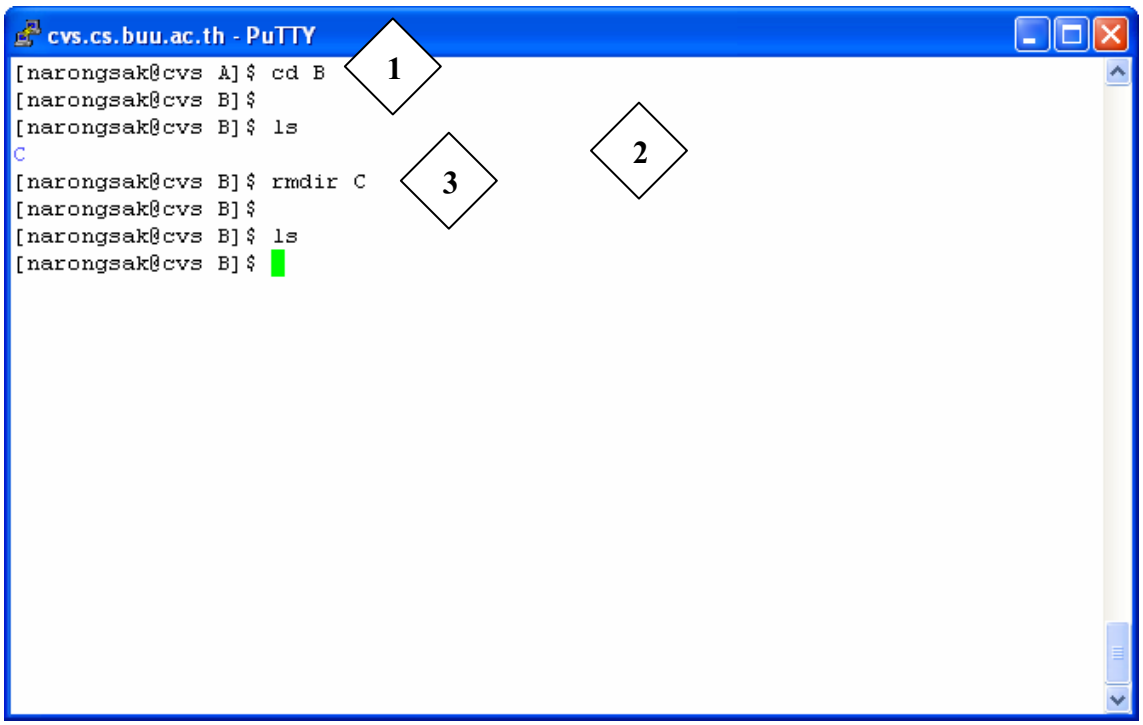
จากรูปที่ 3-11 จากตัวอย่างนี้เป็นการสร้างไดเรคทอรีใหม่ ชื่อ unix ภายใต้ไดเรคทอรี narongsakโดยตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง ls โดยจะแสดงไฟล์และไดเรคทอรีทั้งหมดในไดเรคทอรีปัจจุบันดังตำแหน่งที่ 2 ทั้งนี้ต้องการสร้างไดเรคทอรี unix สามารถทำได้โดยใช้คำสั่ง mkdir unix ดังตำแหน่งที่ 3 ซึ่งเมื่อใช้คำสั่ง ls อีกครั้งจะปรากฏว่า ไดเรคทอรี unix ได้ถูกสร้างเรียบร้อยแล้ว

	คำสั่งในการจัดการไดเรคทอรี : คำสั่ง rmdir
Syntax	**rmdir [OPTION] [FILE]**
Example	**rmdir C**

เป็นคำสั่งที่ใช้สำหรับการลบไดเรคทอรี(ในทำนองเดียวกับ dos) มาจากคำว่า remove ไดเรคทอรี โดย option ที่มักใช้กันใน rmdir คือ

-p จะทำการลบ Child และ Parent ไดเรคทอรีตามลำดับ

สมมติว่า เราต้องการลบไดเรคทอรีชื่อ / A / B / C ถ้าหากไดเรคทอรีปัจจุบันคือ / A / B เราจะต้องทำตามขั้นตอนต่อไปนี้



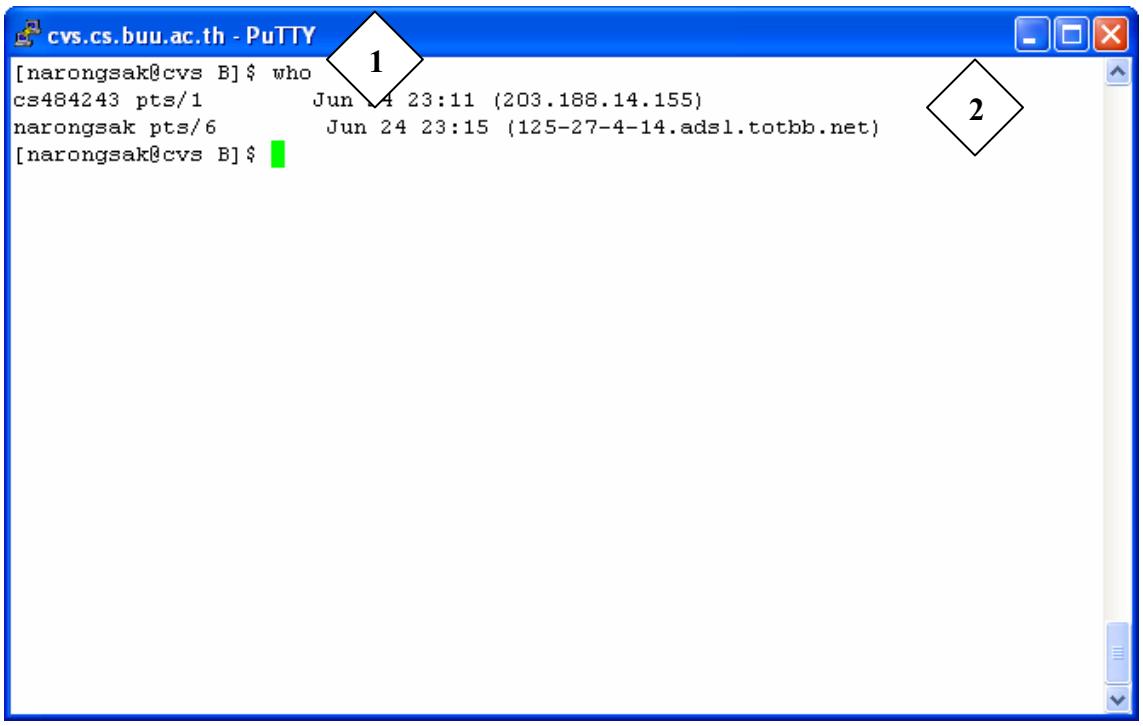
รูปที่ 3-12 หน้าจอแสดงการใช้งานคำสั่ง rmdir

จากรูปที่ 3-11 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง cd B เป็นการเข้าไดเรคทอรี B ดังตำแหน่งที่ 2 ทั้งนี้ต้องการลบไดเรคทอรี c สามารถทำได้โดยใช้คำสั่ง rmdir c ดังตำแหน่งที่ 3 ซึ่งเมื่อใช้คำสั่ง ls อีกครั้งจะปรากฏว่า ไดเรคทอรี c ได้ถูกลบเรียบร้อยแล้ว ซึ่งจากตัวอย่างนี้

คำสั่งในการจัดการเกี่ยวกับผู้ใช้

	คำสั่งในการจัดการเกี่ยวกับผู้ใช้ : คำสั่ง who
Syntax	**who [OPTION] [FILE | ARG1 ARG2]**
Example	**who**

คำสั่ง who ใช้เพื่อแสดงว่ามีใครบ้างที่ login อยู่ในระบบ

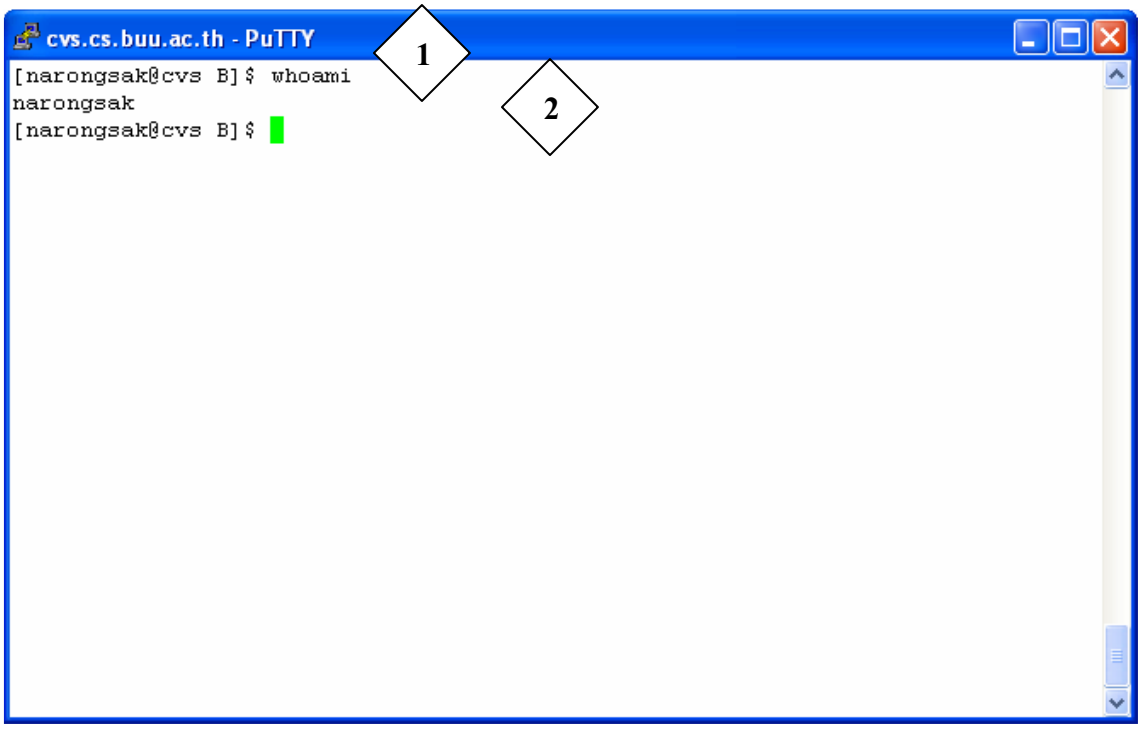


รูปที่ 3-13 หน้าจอแสดงการใช้งานคำสั่ง who

จากรูปที่ 3-13 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง who เป็นการแสดงชื่อของผู้ที่ Login อยู่ในระบบ ดังตำแหน่งที่ 2 จากตัวอย่างนี้ แสดงให้เห็นว่ามี user ที่ชื่อ cs484243 และ narongsak กำลัง Login อยู่ในระบบ

	คำสั่งในการจัดการเกี่ยวกับผู้ใช้ : คำสั่ง whoami
Syntax	**whoami [OPTION]**
Example	**whoami**

คำสั่ง who am i ใช้เพื่อแสดงว่าผู้ใช้ซึ่ง login เข้าสู่ระบบนั้น login ด้วยชื่ออะไร

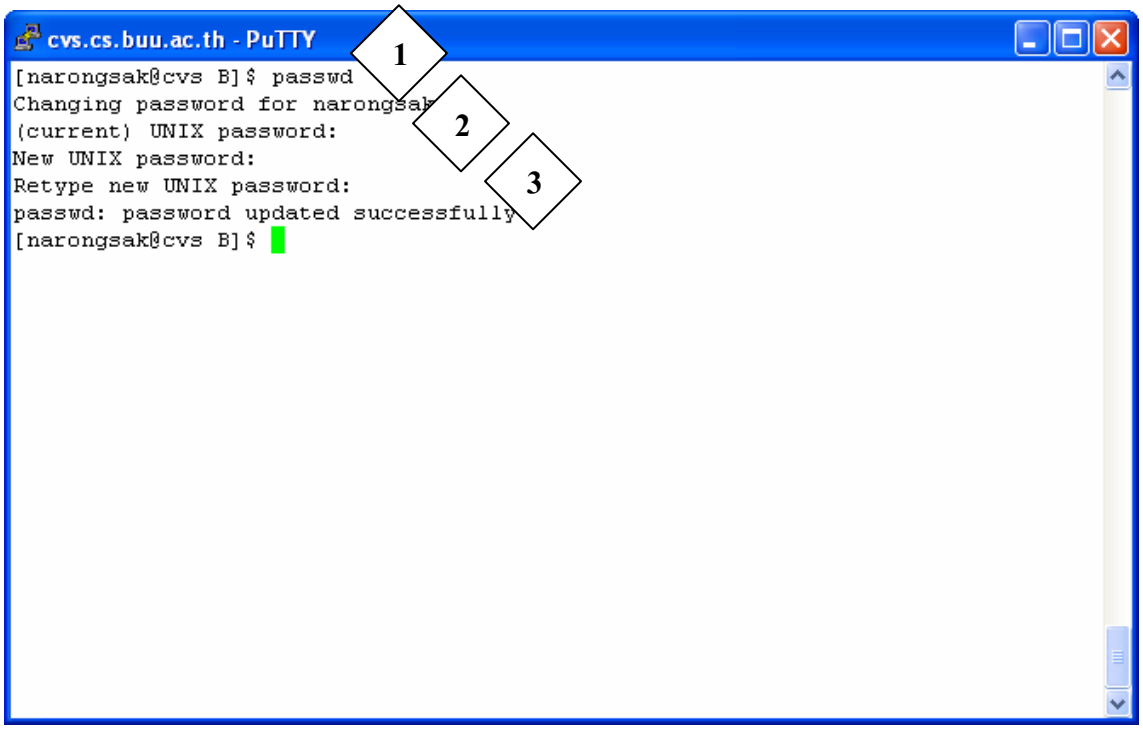


รูปที่ 3-14 หน้าจอแสดงการใช้งานคำสั่ง who

จากรูปที่ 3-14 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง who am i เป็นการแสดงชื่อของผู้ใช้ ดังตำแหน่งที่ 2 จากตัวอย่างนี้ แสดงให้เห็นว่าผู้ใช้ ใช้ชื่อ narongsak ในการ Login เข้าสู่ระบบ

	คำสั่งในการจัดการเกี่ยวกับผู้ใช้ : คำสั่ง passwd
Syntax	**passwd [OPTIONS] [LOGIN]**
Example	**passwd**

คำสั่ง passwd เป็นคำสั่งที่ใช้ในการเปลี่ยน password เมื่อเราใช้คำสั่งนี้ ระบบจะทำการถาม password เดิมก่อน แล้วจึงถาม password ใหม่ และให้ยืนยัน password อีกครั้ง



รูปที่ 3-15 หน้าจอแสดงการใช้งานคำสั่ง passwd

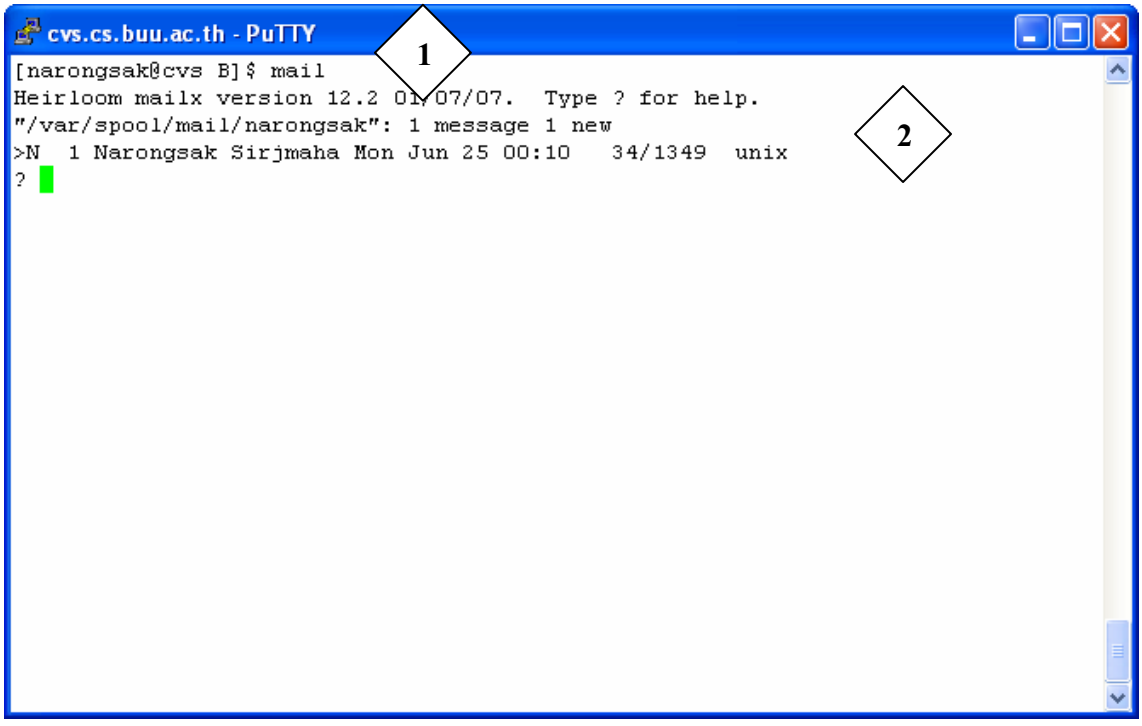
จากรูปที่ 3-15 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง passwd โดยระบบจะถามรหัสเก่าก่อน ดังตำแหน่งที่ 2 จากนั้นระบบจะให้ใส่รหัสใหม่อีก 2 ครั้ง ดังตำแหน่งที่ 3 รหัสใหม่ต้องใส่เหมือนกันทั้งสองครั้ง จึงจะสามารถเปลี่ยนรหัสใหม่ได้

คำสั่งในการจัดการการติดต่อสื่อสาร

	คำสั่งในการจัดการติดต่อสื่อสาร : คำสั่ง mail
Syntax	**mail**
Example	**mail**

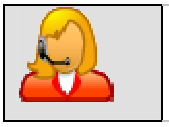
คำสั่ง mail จะเป็นคำสั่งที่ใช้ในการส่ง mail หรือ อ่าน mail ที่อยู่ในระบบยูนิกซ์เท่านั้น เมื่อใช้คำสั่งแล้ว ระบบจะขึ้น prompt (เครื่องหมาย ?) เราสามารถใส่คำสั่งลงไปได้ เช่น

h	แสดง mail ที่มีอยู่
หมายเลข	ใช้สำหรับเลือก mail ที่เราต้องการอ่าน
d หมายเลข	ใช้สำหรับลบ mail
q	ใช้สำหรับการจบการทำงานและกลับไปสู่ UNIX prompt



รูปที่ 3-16 หน้าจอแสดงการใช้งานคำสั่ง mail

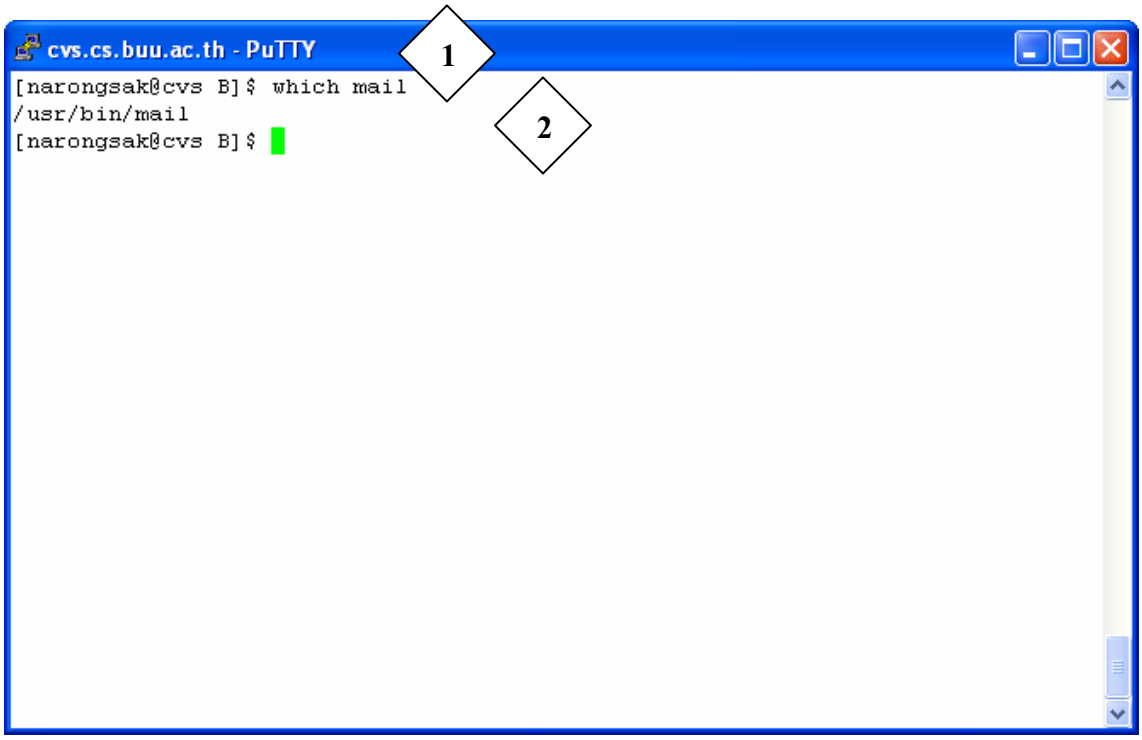
จากรูปที่ 3-16 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง mail โดยระบบจะแสดงว่ามีเมล์กี่ฉบับที่ถูกส่งมา ส่งมาเมื่อไหร่ เวลาเท่าใด ดังตำแหน่งที่ 2



คำสั่งในการค้นหาไฟล์ และระบบรักษาความปลอดภัย

	คำสั่งในการค้นหาไฟล์ และระบบรักษาความปลอดภัย: คำสั่ง which
Syntax	**which [OPTION] [--] PROG_NAME [...]**
Example	**which mail**

คำสั่ง which เป็นคำสั่งที่ใช้แสดงที่อยู่ของไฟล์



รูปที่ 3-17 หน้าจอแสดงการใช้งานคำสั่ง mail

จากรูปที่ 3-17 ตำแหน่งที่ 1 เมื่อพิมพ์คำสั่ง which mail โดยระบบจะแสดงที่อยู่ของไฟล์ว่าอยู่ที่ใด ดังตำแหน่งที่ 2 จากตัวอย่างระบบจะแสดงที่อยู่ของไฟล์ mail ให้เรารู้ว่าอยู่ในไดเรคทอรีไหน

	คำสั่งในการค้นหาไฟล์ และระบบรักษาความปลอดภัย: คำสั่ง chmod
Syntax 1	**chmod [OPTION]... MODE[,MODE]... FILE...**
Syntax 2	**chmod [OPTION]... OCTAL-MODE FILE...**
Syntax 3	**chmod [OPTION]... --reference=RFILE FILE...**
Example	**chmod o+w test.txt**

คำสั่ง chmod เป็นคำสั่งที่ใช้สำหรับกำหนดสิทธิของผู้ใช้ในกลุ่มต่างๆ ให้สามารถเข้าถึงไฟล์ต่างๆได้ ระบบจะแบ่งกลุ่มผู้ใช้ออกเป็น 3 กลุ่มด้วยกัน

1. เจ้าของ (owner) คือผู้ที่สร้างไฟล์ขึ้นมา แทนด้วยตัวอักษร (u)
2. กลุ่ม (group) คือ กลุ่มผู้ใช้ ผู้ใช้ที่อยู่กลุ่มเดียวกันสามารถใช้งานไฟล์ของเพื่อนร่วมกลุ่มได้ แทนด้วยตัวอักษร (g)
3. ผู้อื่น (other) คือ บุคคลที่ไม่ได้อยู่ในกลุ่ม หรือ ไม่ใช่เจ้าของไฟล์ แทนด้วยตัวอักษร (o)

ผู้ใช้สามารถกำหนดสิทธิต่างๆให้กับไฟล์ได้ โดยแบ่งออกเป็น 3 อย่างด้วยกัน

1. read คือสิทธิในการอ่านไฟล์ แทนด้วยตัวอักษร (r)
2. write คือ สิทธิในการแก้ไขไฟล์ แทนด้วยตัวอักษร (w)
3. execute คือ สิทธิในการใช้งานไฟล์ แทนด้วยตัวอักษร (x)

รายละเอียดของไฟล์สามารถดูได้ด้วยคำสั่ง ls –l

```
drwxr-xr-x  2 narongsak staff     4096 May 30 15:19 17-3-50
drwxr-xr-x  2 narongsak staff     4096 Dec 29 17:03 26-15-49
drwxr-xr-x  3 narongsak staff     4096 Jun 24 23:35 A
-rw-r--r--  1 narongsak staff    32256 Jun 23 19:43 BLS.doc
```

ตัวอักษรสามารถแบ่งได้ดังนี้

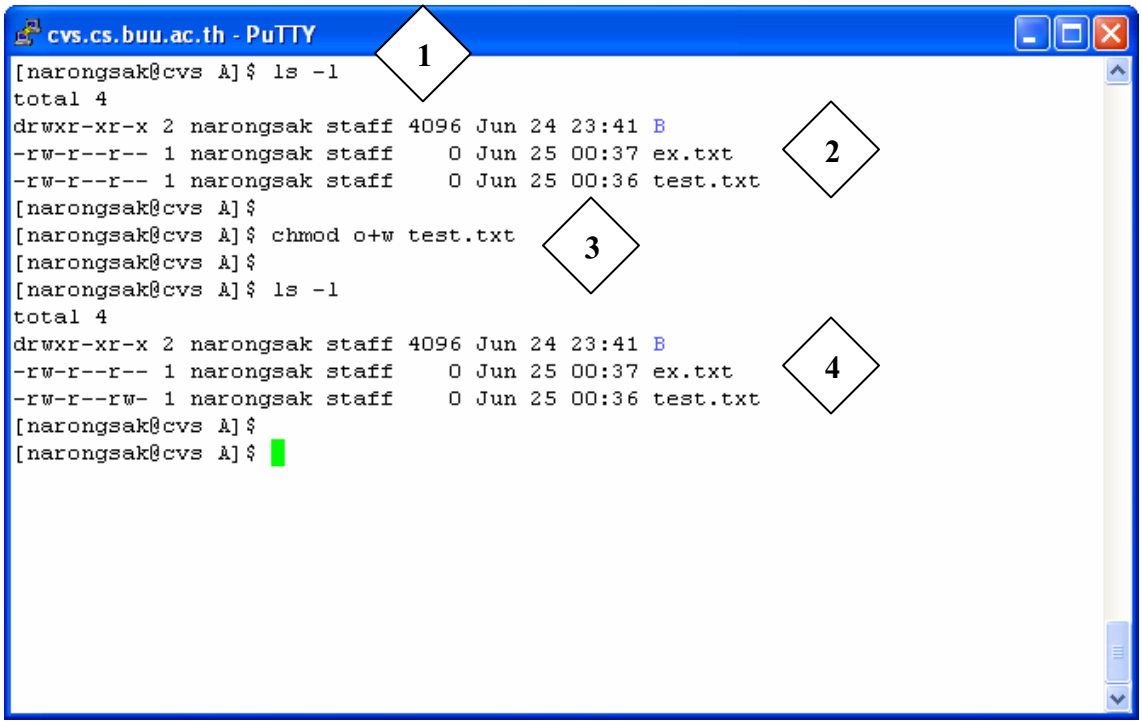
ตัวที่ 1 บอกว่าเป็นไดเรคทอรีหรือไม่ ถ้าเป็นจะมี ตัวอักษร d ถ้าไม่เป็นจะมีเครื่องหมาย (-)

ตัวที่ 2-4 บอกว่าเป็นสิทธิของเจ้าของไฟล์

ตัวที่ 5-7 บอกว่าเป็นสิทธิของผู้ใช้ที่อยู่ในกลุ่มเดียวกัน

ตัวที่ 8-10 บอกว่าเป็นสิทธิของผู้ใช้อื่นๆนอกเหนือจาก 2 กลุ่มแรก

รูปแบบการใช้คำสั่ง chmod [u / g / o] [+ / -] [r / w / x] filename



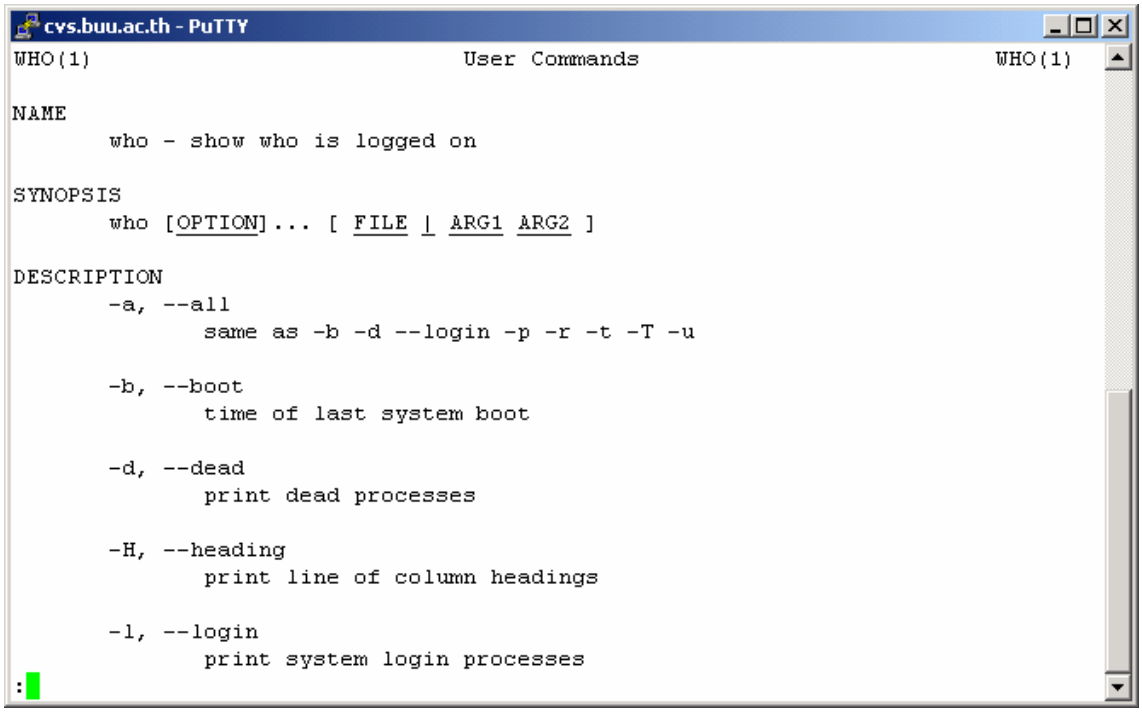
รูปที่ 3-18 หน้าจอแสดงการใช้งานคำสั่ง chmod

จากรูปที่ 3-18 เมื่อเราต้องการที่จะเปลี่ยนสิทธิ์ให้กับไฟล์ที่ต้องการ โดยตำแหน่งที่ 1 พิมพ์คำสั่ง ls –l ซึ่งแสดงข้อมูลเกี่ยวกับแฟ้ม และไดเรคทอรีในไดเรคทอรีปัจจุบัน โดยแสดงหนึ่งบรรทัดต่อหนึ่งไฟล์หรือไดเรคทอรีแบบเต็มรูปแบบ(long format) ดังตำแหน่งที่ 2 ทั้งนี้ต้องการแก้ไขไฟล์ที่ชื่อ test.txt ถูกกำหนดสิทธิให้ผู้ใช้อื่นสามารถแก้ไขไฟล์ได้ โดยพิมพ์คำสั่ง chmod ดังตำแหน่งที่ 3 โดย จากนั้นคำสั่ง ls –l จะเห็นได้ว่าไฟล์ที่ชื่อ test.txt ถูกกำหนดสิทธิให้ผู้ใช้อื่นสามารถแก้ไขไฟล์ได้ซึ่งจากเดิมทำไม่ได้ดังตำแหน่งที่ 4

คำสั่งพื้นฐานอื่น ๆ

	คำสั่งในการจัดการไฟล์ : คำสั่ง man
Syntax	**man [COMMAND]**
Example	**man who**

คำสั่ง man เป็นคำสั่งแสดงข้อความ ที่อธิบายการใช้งานของคำสั่ง รายละเอียดต่างๆ รวมถึง option ที่คำสั่งสามารถใช้ได้ด้วย

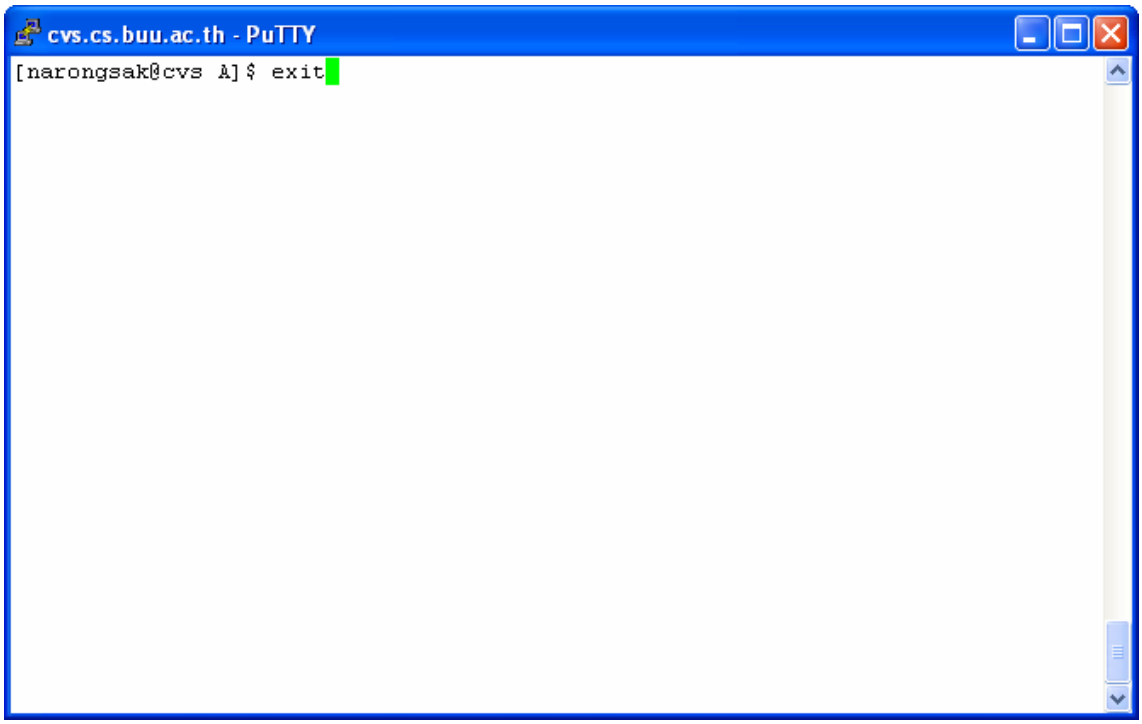


รูปที่ 3-19 หน้าจอแสดงการใช้งานคำสั่ง man

จากตัวอย่างข้างต้น แสดงรายละเอียดของคำสั่ง clear และเมื่อต้องการจะออก ให้กด q

	คำสั่งในการจัดการไฟล์ : คำสั่ง exit
Syntax	**exit**
Example	**exit**

คำสั่ง exit เป็นคำสั่งสำหรับออกจากระบบ UNIX



รูปที่ 3-20 หน้าจอแสดงการใช้งานคำสั่ง exit

เมื่อใช้คำสั่งนี้แล้วหน้าจอจะว่างเปล่า จะใช้คำสั่งนี้ก็ต่อเมื่อทำงานในระบบเสร็จเรียบร้อยแล้วและไม่มีงานที่จะทำต่อแล้ว

บทที่ 4

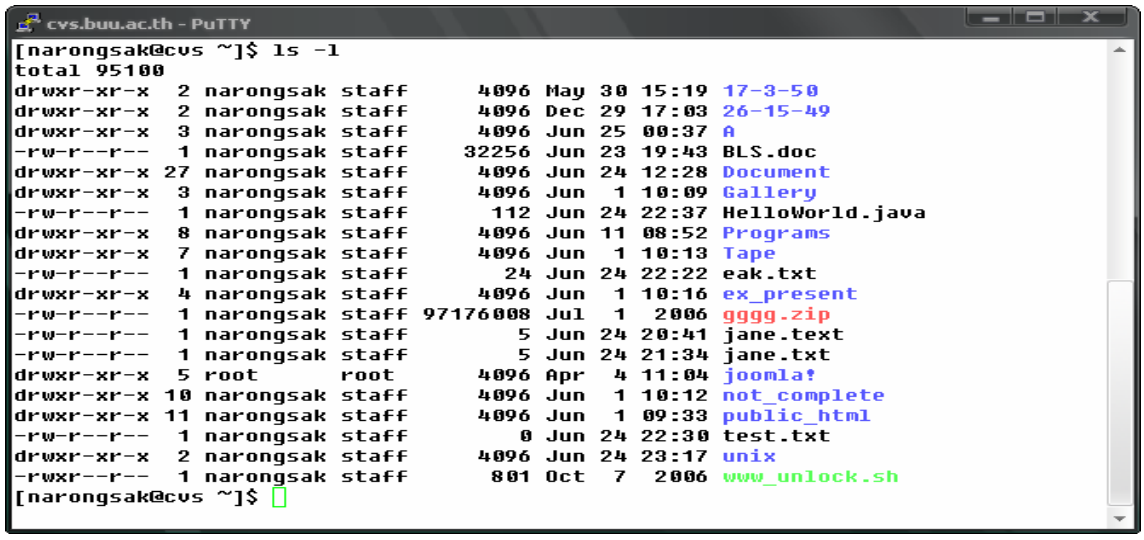
สิทธิการเข้าถึงของระบบแฟ้มข้อมูล (Permission)

สิทธิการเข้าถึงของระบบแฟ้มข้อมูล (Permission)

เนื่องจากลีนุกซ์มีระบบการทำงานแบบผู้ใช้หลายคน (Multiuser) ดังนั้นจึงมีความจำเป็นที่ต้องมีการกำหนดสิทธิในการเข้าถึงแฟ้มข้อมูลและไดเรคทอรีต่างๆ โดย จะแบ่งเป็น 3 กลุ่มคือ

- คนที่เป็นเจ้าของไฟล์ (User)
- คนที่อยู่ในกลุ่มเดียวกับเจ้าของไฟล์ (Group)
- คนที่อยู่คนละกลุ่มกับเจ้าของไฟล์ (Other)

เมื่อใช้คำสั่ง ls -l จะได้ผลลัพธ์ออกมาดังนี้



รูปที่ 4-1 ผลลัพธ์จากการใช้คำสั่ง ls -l

```
-rw-r--r--  1 narongsak staff    32256 Jun 23 19:43 BLS.doc
drwxr-xr-x 27 narongsak staff     4096 Jun 24 12:28 Document
drwxr-xr-x  3 narongsak staff     4096 Jun  1 10:09 Gallery
-rw-r--r--  1 narongsak staff      112 Jun 24 22:37 HelloWorld.java
```

ให้สังเกตข้อมูลจากกลุ่มแรก ประกอบด้วยตัวอักษร คือ r w x d - ความหมายของตัวอักษรเหล่านี้ คือ

r ย่อมาจาก read หมายถึง สามารถอ่านไฟล์ หรือไดเรคทอรีได้

w ย่อมาจาก write หมายถึง สามารถเขียนข้อมูลลงไปในไฟล์ หรือสามารถที่จะกระทำการต่างๆกับไดเรคทอรีได้

x ย่อมาจาก execute หมายถึงสามารถที่จะประมวลผลไฟล์ได้ หรือสามารถที่จะเข้าไปในไดเรคทอรีได้

d เป็นสัญลักษณ์เพื่อบอกให้รู้ว่าเป็นไดเรคทอรี

- เป็นสัญลักษณ์เพื่อบอกให้รู้ว่าเป็นไฟล์

สิทธิ	เกี่ยวกับไฟล์	เกี่ยวกับไดเรคทอรี
read	ดูเนื้อหา อ่านข้อมูลได้	ดูข้อมูลหรือรายชื่อได้
write	เปลี่ยนแปลง แก้ไขเนื้อหาได้	สร้างหรือลบไดเรคทอรีได้
execute	สั่ง execute (ประมวลผลคำสั่ง)ได้	เปลี่ยน ค้นหา สำเนาจากไฟล์ในไดเรคทอรีนั้นได้

ตารางที่ 4-1 แสดงสิทธิของไฟล์และไดเรคทอรี

ค่าของสิทธิไฟล์หนึ่งมีค่าเป็น - rwx r-x r--

TYPE	USER (u)			GROUP (g)			OTHER (o)		
-	r	w	x	r	-	x	r	-	-
	4	2	1	4	0	1	4	0	0

ตารางที่ 4-2 แสดงสิทธิของไฟล์และไดเรคทอรีแบบแสดงรายเอียดของ - rwx r-x r--

สิทธิตามตารางที่ 4-2 หมายถึง

ตัวแรกเป็น – บอกให้ทาบว่าข้อมูลนี้คือไฟล์ ไม่ใช่ไดเรคทอรี

3 ตัวอักษร USER (u) คือ สิทธิที่เจ้าของดำเนินการได้ คือ r w x อ่าน เขียน ประมวลผลได้

3 ตัวอักษร GROUP (g) คือ สิทธิที่คนที่อยู่กลุ่มเดียวกับเจ้าของทำได้ คือ r x อ่าน ประมวลผลได้

3 ตัวอักษร OTHER (o) คือ สิทธิที่คนที่อยู่ต่างกลุ่มกับเจ้าของทำได้ คือ r อ่านได้

ถ้าคิดเป็นตัวเลขจะได้ค่าตัวเลขของสิทธิโดยให้นำค่าตัวเลขมารวมกันจะได้เป็น 754

ตัวอย่างรูปแบบของสิทธิ

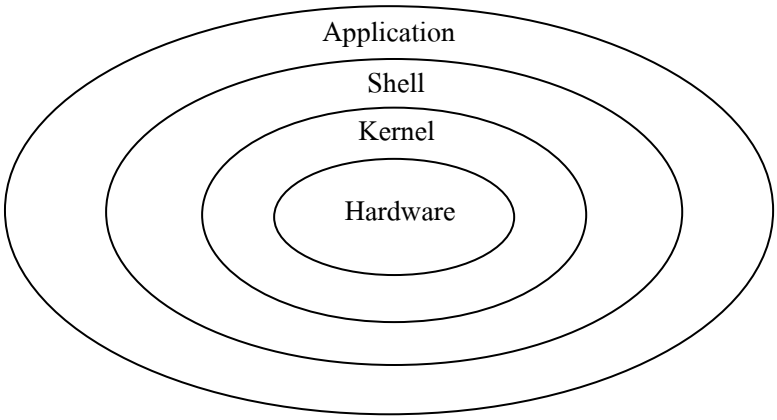
- r w x --- ---	700	หมายถึง ไฟล์ที่ เจ้าของอ่าน เขียน ประมวลผลได้
d r w - r - - r - x	645	หมายถึง ไดเรคทอรีที่เจ้าของอ่านเขียนได้ กลุ่มเดียวกันอ่านได้ กลุ่มอื่นอ่านและประมวลผลได้
- r - - r - - r - -	444	หมายถึง ไฟล์ที่ทุกกลุ่มอ่านได้อย่างเดียว

บทที่ 5

Shell เบื้องต้น

โครงสร้างพื้นฐานการทำงานของระบบยูนิกซ์

โครงสร้างพื้นฐานการทำงานของระบบยูนิกซ์มีอยู่ 4 ส่วนด้วยกัน คือ ฮาร์ดแวร์ (Hardware), เคอร์เนล (Kernel), เชลล์ (Shell) และแอพพลิเคชัน (Application) ดังรูปที่ 5-1



รูปที่ 5-1 แสดงโครงสร้างพื้นฐานการทำงานของระบบยูนิกซ์

เชลล์ (Shell) คือ โปรแกรมหนึ่งบนระบบยูนิกซ์ที่ทำหน้าที่เป็นส่วนติดต่อผู้ใช้ (interface) ระหว่างผู้ใช้กับระบบปฏิบัติการยูนิกซ์ (เคอร์เนล) ซึ่งเชลล์ไม่ได้เป็นส่วนหนึ่งของเคอร์เนล แต่ใช้เคอร์เนลในการประมวลผล ผู้ใช้สามารถสั่งงานระบบปฏิบัติการได้โดยผ่านทางเชลล์เท่านั้น โปรแกรมเชลล์ยังมีคุณสมบัติของ Shell Programming Language ทำให้ผู้ใช้สามารถนำคำสั่งต่างๆ ของเชลล์มาเขียนเป็นโปรแกรมเก็บเป็นไฟล์ไว้ได้ เรียกว่า Shell Script

ประเภทของเชลล์ที่นิยมใช้ในปัจจุบัน

- **Bourne shell** (/bin/sh) เป็นเชลล์ในยุคแรกๆ ที่มีใช้กันอย่างแพร่หลาย มีการกำหนดโครงสร้างภาษาคล้ายๆ กับภาษาอัลกอ (Algo Language) สามารถเขียนเป็น shell script ได้ และยังเป็น เชลล์มาตราฐานที่มีในระบบปฏิบัติการยูนิกซ์ทุกตัว และยังสามารถย้าย shell script ไปยังยูนิกซ์ระบบอื่นโดยไม่ต้องแก้ไขอะไรได้อีกด้วย จะมี default prompt เป็นเครื่องหมาย “ $ ”
- **C shell** (/bin/csh) เป็นเชลล์ที่พัฒนาขึ้นมาหลังจาก Bourne shell มีรูปแบบคำสั่งและไวยากรณ์เหมือนกับภาษาซี (C Language) มีฟังก์ชันการทำงานหลายที่ดีและอย่างสะดวก อีกทั้งยังสามารถควบคุมการไหลของข้อมูลได้ดีกว่า Bourne shell และยังมีความสามารถในการเรียกใช้คำสั่งที่ใช้ไปแล้ว จะมี default prompt เป็นเครื่องหมาย “ % ”
- **Korn shell** (/bin/ksh) เป็น shell ที่พัฒนามาจาดต้นแบบของ Bourne shell และ C shell สามารถทำงานใน function ของ Bourne shell ได้ทุกอย่าง การเขียน shell script ทำได้ง่ายและรัดกุมขึ้น สามารถนำคำสั่งที่ใช้ไปแล้วกลับมา execute ไปใหม่ได้ ถือได้ว่า Korn shell เป็นการรวมเอาข้อดีของ Bourne shell และ C shell มาไว้ด้วยกัน แต่ไม่ได้มีใน UNIX ทุกตัว จะมี default prompt เป็นเครื่องหมาย “ $ ”
- **Bourne again shell** (/bin/bash หรือ /usr/local/bin/bash) เป็นการเอา Bournr shell นำกลับมาพัฒนาใหม่ สามารถทำงานแบบ line editing ได้ และยังได้เพิ่มประสิทธิภาพในการทำงานอีกหลายอย่าง bash shell นี้ไม่ใช่ standard UNIX shell แต่เป็น default shell ของ linux ในปัจจุบัน จะมี default prompt เป็นเครื่องหมาย “ $ ”

การแสดงประเภทของเชลล์ทั้งหมดในระบบปฏิบัติของคุณ

ผู้ใช้สามารถตรวจสอบดูประเภทของเชลล์ทั้งหมดที่ระบบปฏิบัติการมี ได้โดยใช้คำสั่ง

```
$ cat /etc/shells
```

การแสดงประเภทของเชลล์ที่กำลังใช้อยู่ในขณะนั้น

ผู้ใช้สามารถตรวจสอบดูประเภทของเชลล์ที่กำลังใช้งานอยู่ในขณะนั้น ได้โดยใช้คำสั่ง

```
$ echo $SHELL
```

	ทำไมจึงต้องเขียน Shell Script

เหตุผลที่ต้องเขียน Shell Script นั้น ปกติแล้วจะขึ้นอยู่กับผู้ใช้งานแต่ละคน แต่โดยส่วนมากแล้วนั้นจะมีเหตุผลในการเขียนดังนี้

- Shell script สามารถรับข้อมูลเข้า (Input) จากผู้ใช้ ไฟล์ และแสดงผลลัพธ์ออกทางหน้าจอได้เหมือนกับการใช้คำสั่ง (Command) โดยตรง
- มีประโยชน์อย่างมากในการสร้างคำสั่งที่เป็นส่วนบุคคล
- ลดเวลาในการคีย์คำสั่ง
- สามารถทำให้งานบางอย่างเป็นอัตโนมัติได้
- เป็นส่วนหนึ่งของงานที่เป็นอัตโนมัติของผู้ดูแลระบบ
-

	ขั้นตอนในการเขียน Shell Script

1. ใช้โปรแกรมประเภท editor ในการเขียน Shell Script
2. เพิ่มสิทธิให้ไฟล์ Shell Script ที่เขียนขึ้นนั้นสามารถ execute ได้ด้วยคำสั่ง

   $ chmod +x script

   $ chmod 755 script
3. execute ไฟล์ Shell Script ดังกล่าว ด้วยคำสั่ง

   $ sh script ในกรณีที่เป็น Bourne shell

   $ bash script ในกรณีที่เป็น Bourne again shell

   $ ./script

	ตัวอย่างที่ 1 ของ Shell Script
Example 1	clear echo "Hello $USER" echo "Today is \c ";date echo "Number of user login : \c" ; who | wc -l echo "Calendar" cal exit 0

ผลลัพธ์	Hello nattawut Today is Tue Jul 24 15:25:08 ICT 2007 Number of user login : 1 Calendar July 2007 Su Mo Tu We Th Fr Sa 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

บรรทัดที่ 1 เป็นการเคลียร์หน้าจอแสดงผล

บรรทัดที่ 2 เป็นการแสดงข้อความทักทายผู้ใช้งาน โดยตัวแปร $USER จะเป็นตัวแปรที่ใช้เก็บชื่อผู้ใช้ไว้

บรรทัดที่ 3 เป็นการแสดงวันที่และเวลาปัจจุบัน

บรรทัดที่ 4 เป็นการแสดงจำนวนผู้ใช้งานในระบบ ณ เวลานั้น ๆ

บรรทัดที่ 5-6 เป็นการแสดงปฏิทินของเดือนปัจจุบัน

	ตัวอย่างที่ 2 ของ Shell Script
Example	# # Script to read your name from key-board # echo "Your Name please:" read name echo "Hello $name, Lets be friend!"

ผลลัพธ์	Your Name please: พิมพ์ข้อมูล Hello ข้อมูลที่พิมพ์เข้าไป, Lets be friend!

บรรทัดที่ 1 เป็นการแสดงข้อความเพื่อให้ผู้ใช้กรอกชื่อของตน

บรรทัดที่ 2 เป็นการรับข้อมูลเข้าจากผู้ใช้ผ่านทางคีย์บอร์ด แล้วเก็บไว้ในตัวแปรชื่อว่า name

บรรทัดที่ 3 เป็นการแสดงค่าที่อยู่ในตัวแปร name

ข้อสังเกต

การกำหนดค่าให้กับตัวแปรนั้น สามารถใช้ชื่อตัวแปรในการดำเนินการได้เลย แต่ในส่วนของการแสดงค่าจากตัวแปรนั้นจะต้องใส่เครื่องหมาย $ นำหน้าตัวแปรนั้น ๆ ด้วย

	ตัวอย่างที่ 3 ของ Shell Script
Example	#!/bin/sh # # Script that demos, command line args # echo "Total number of command line argument are $#" echo "$0 is script name" echo "$1 is first argument" echo "$2 is second argument"
ผลลัพธ์	ผลลัพธ์หลังการรันด้วย ./script A B Total number of command line argument are 2 **script** is script name A is first argument B is second argument

Shell Script ข้างต้นเป็นตัวอย่างในการรับค่าพารามิเตอร์ผ่านทาง command line โดยพารามิเตอร์ที่ผู้ใช้ระบุนั้นจะถูกเก็บอยู่ในตัวแปร $1, $2 ,... ตามลำดับ ส่วนจำนวนของตัวแปรที่ระบุเข้ามา จะถูกเก็บอยู่ในตัวแปร $#

การเปลี่ยนทิศทางของข้อมูลเข้าและผลลัพธ์ที่ได้จากการรัน Shell Script

นอกจากการรับข้อมูลเข้าผ่านทางคีย์บอร์ดนั้น (ตามตัวอย่างที่ 2) และการแสดงผลลัพธ์ที่ได้ออกทางหน้าจอนั้น ยังมีรูปแบบของการรับข้อมูลเข้าและแสดงผลลัพธ์ในรูปแบบอื่นอีกด้วย

	1. การเปลี่ยนทิศทางแบบ >
เป็นการเปลี่ยนทิศทางในการแสดงผลจากหน้าจอลงไฟล์ โดยมีรูปแบบของการใช้งาน ดังนี้	
Example	./program > output_file
ผลลัพธ์	ผลลัพธ์ที่ได้จากการรันโปรแกรม จะถูกเขียนลงไฟล์ที่ชื่อว่า output_file โดยไฟล์ดังกล่าวนั้นไม่ว่าจะมีข้อมูลเดิมอยู่หรือไม่ก็ตาม จะถูกเขียนทับด้วยผลลัพธ์จากการรันโปรแกรมทั้งหมด

	2. การเปลี่ยนทิศทางแบบ >>
เป็นการเปลี่ยนทิศทางในการแสดงผลจากหน้าจอลงไฟล์ โดยมีรูปแบบของการใช้งาน ดังนี้	
Example	./program >> output_file
ผลลัพธ์	ผลลัพธ์ที่ได้จากการรันโปรแกรม จะถูกเขียนลงไฟล์ที่ชื่อว่า output_file โดยจะเป็นการเขียนต่อไฟล์ กล่าวคือข้อมูลเดิมในไฟล์ output_file ยังคงอยู่และมีผลจากการรันโปรแกรมต่อท้ายข้อมูลนั้น

	3. การเปลี่ยนทิศทางแบบ <
เป็นการเปลี่ยนทิศทางในรับข้อมูลเข้าจากไฟล์แทนการรับด้วยคีย์บอร์ด โดยมีรูปแบบของการใช้งาน ดังนี้	
Example	./program < input_file

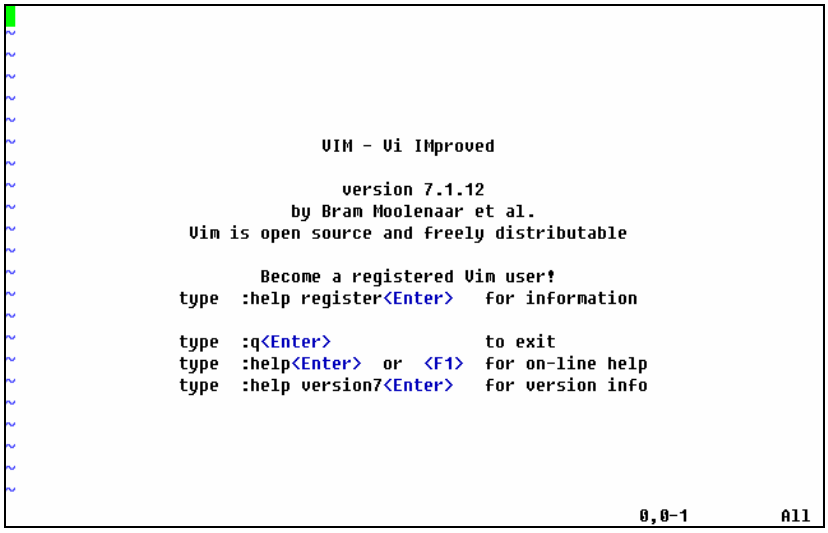
	4. การนำเอาผลลัพธ์ของโปรแกรมหนึ่งไปเป็นข้อมูลเข้าสำหรับโปรแกรมที่สองด้วย |
เราสามารถนำเอาผลลัพธ์ที่ได้จากการรันโปรแกรมที่ 1 ไปเป็นข้อมูลเข้าสำหรับโปรแกรมที่ 2 ได้โดยมีรูปแบบของการใช้งาน ดังนี้	
Example	./program1 | ./program2
ผลลัพธ์	ผลลัพธ์ที่ได้จากการรันโปรแกรม จะถูกเขียนลงไฟล์ที่ชื่อว่า output_file โดยจะเป็นการเขียนต่อไฟล์ กล่าวคือข้อมูลเดิมในไฟล์ output_file ยังคงอยู่และมีผลจากการรันโปรแกรมต่อท้ายข้อมูลนั้น

	การใช้งานของการเปลี่ยนทิศทางทั้ง 4 รูปแบบนั้นสามารถใช้งานร่วมกันได้ เช่น
เราสามารถนำเอาผลลัพธ์ที่ได้จากการรันโปรแกรมที่ 1 ไปเป็นข้อมูลเข้าสำหรับโปรแกรมที่ 2 ได้โดยมีรูปแบบของการใช้งาน ดังนี้	
Example	./program < input_file > output_file
คำอธิบาย	จากคำสั่งข้างต้นเป็นการรันโปรแกรมโดยรับข้อมูลเข้าจากไฟล์ input_file และบันทึกผลจากการรันโปรแกรมลงในไฟล์ output_file

บทที่ 6

การใช้งานโปรแกรม vi เบื้องต้น

หน้าจอโปรแกรม vi



รูปที่ 6-1 หน้าจอโปรแกรม vi

การทำงานของโปรแกรม vi

โปรแกรม vi แบ่งการทำงานออกเป็น 3 โหมด คือ

1. โหมดคำสั่ง (Command Mode)
2. โหมดพิมพ์ข้อความ (Insert Mode)
3. โหมดคำสั่งบรรทัดล่างสุด (Lastline Mode)

แต่ละโหมดของโปรแกรมจะมีรูปแบบการทำงานและหน้าที่เฉพาะที่แตกต่างกัน ได้แก่

1. **โหมดคำสั่ง** เป็นโหมดที่ใช้ในการสั่งให้โปรแกรมทำงานต่าง ๆ ประกอบด้วย
    - คำสั่งในการเลื่อนเคอเซอร์และหน้าจอ
    - คำสั่งในการเข้าสู่โหมดพิมพ์ข้อความ
    - คำสั่งในการลบตัวอักษร
    - คำสั่งในการลบบรรทัด
    - คำสั่งในการแทนที่ตัวอักษร
    - คำสั่งในการเพิ่มบรรทัดใหม่
    - คำสั่งในการยกเลิกการกระทำล่าสุด

    เป็นต้น
2. **โหมดพิมพ์ข้อความ** เป็นโหมดที่ใช้ในการพิมพ์หรือป้อนข้อความบนพื้นที่ทำงานของโปรแกรม
3. **โหมดคำสั่งบรรทัดล่างสุด** เป็นโหมดที่ใช้ในการสั่งงานโปรแกรมทำงานดังนี้
    - คำสั่งที่เกี่ยวกับการบันทึกข้อมูลลงไฟล์
    - คำสั่งในการค้นหาข้อความ

การเริ่มต้นใช้งานโปรแกรม vi

การเข้าสู่โปรแกรม vi นั้นสามารถใช้ได้โดยการพิมพ์คำสั่ง

vi <ชื่อแฟ้ม>

คำสั่งในการเลื่อนเคอเซอร์และหน้าจอขณะอยู่ในโหมดคำสั่ง

ในขณะที่อยู่ในโหมดคำสั่ง ผู้ใช้สามารถเลื่อนเคอเซอร์ไปยังตำแหน่งของอักขระใด ๆ บนหน้าจอได้ และสามารถเลื่อนหน้าจอเพื่อแสดงข้อมูลที่ส่วนใดส่วนหนึ่งของหน้าจอโปรแกรมได้โดยใช้คำสั่งในการเลื่อนเคอเซอร์ ดังตารางที่ 5-1

เลื่อนเคอเซอร์ไปทางขวา 1 ตัวอักษร	**l**
เลื่อนเคอเซอร์ไปทางซ้าย 1 ตัวอักษร	**h**
เลื่อนเคอเซอร์ไปทางขวา 1 คำ	**w**
เลื่อนเคอเซอร์ไปทางซ้าย 1 คำ	**b**
เลื่อนเคอเซอร์ขึ้นบน 1 บรรทัด	**k**
เลื่อนเคอเซอร์ลงล่าง 1 บรรทัด	**j**
เลื่อนเคอเซอร์ไปต้นบรรทัด	**0 (ศูนย์)**
เลื่อนเคอเซอร์ไปท้ายบรรทัด	**$**
เลื่อนเคอเซอร์ไปทางขวา 1 ประโยค	**)**
เลื่อนเคอเซอร์ไปทางซ้าย 1 ประโยค	**(**
เลื่อนเคอเซอร์ไปย่อหน้าถัดไป	**}**
เลื่อนเคอเซอร์ไปย่อหน้าก่อน	**{**
เลื่อนเคอเซอร์ไปยังมุมบนซ้ายของจอ	**H**
เลื่อนเคอเซอร์ไปยังบรรทัดที่อยู่กึ่งกลางจอ	**M**
เลื่อนเคอเซอร์ไปยังบรรทัดล่างสุดของจอ	**L**
เลื่อนหน้าจอขึ้นครึ่งหน้าจอ	**Ctrl-U (up)**
เลื่อนหน้าจอลงครึ่งหน้าจอ	**Ctrl-D (down)**
เลื่อนหน้าจอขึ้นหนึ่งหน้าจอ	**Ctrl-F (forward)**
เลื่อนหน้าจอลงหนึ่งหน้าจอ	**Ctrl-B (backward)**
เลื่อนไปบรรทัดแรกสุด	**gg**
เลื่อนไปบรรทัดท้ายสุด	**G**

ตารางที่ 5-1 แสดงคำสั่งในการเลื่อนเคอเซอร์

การเลื่อนเคอเซอร์ไปยังบรรทัดที่ต้องการทำได้โดยการพิมพ์หมายเลขบรรทัดตามด้วยคำสั่ง **G** หรือ **gg** เช่น 25G หรือ 25gg หมายถึงการเลื่อนเคอเซอร์ไปยังบรรทัดที่ 25

คำสั่งในการเข้าสู่โหมดพิมพ์ข้อความ

คำสั่งแทรก (Insert)

การแทรกที่หน้าตำแหน่งเคอเซอร์ **i**

การแทรกที่ต้นบรรทัด **I**

เมื่อใช้คำสั่ง **i** หรือ **I** แล้ว ผู้ใช้สามารถป้อนข้อความที่ต้องการแทรกได้ทันที เมื่อเสร็จสิ้นการพิมพ์ข้อความที่ต้องการแทรกแล้ว ให้กดแป้น Esc เพื่อกลับสู่โหมดคำสั่ง

คำสั่งพิมพ์ข้อความต่อ (Append)

การพิมพ์ข้อความต่อที่หลังตำแหน่งเคอเซอร์ **a**

การพิมพ์ข้อความต่อที่ท้ายบรรทัด **A**

คำสั่งแทรกบรรทัดใหม่ (Open)

โดยปกติเมื่ออยู่ในโหมดพิมพ์ข้อความเมื่อผู้ใช้พิมพ์ข้อความจนกระทั่งเต็มบรรทัดแล้ว หากผู้ใช้ต้องการขึ้นบรรทัดใหม่สามารถทำได้โดยการกดแป้น Enter อย่างไรก็ตามหากมีการพิมพ์ข้อความไว้แล้ว และต้องการแทรกบรรทัดใหม่ลงระหว่างข้อความเดิมทำได้โดยเลือกใช้คำสั่งดังนี้

แทรกบรรทัดใหม่ใต้บรรทัดเคอเซอร์ **o**

แทรกบรรทัดใหม่เหนือบรรทัดเคอเซอร์ **O**

การใช้คำสั่ง o และ O ในโหมดคำสั่งจะทำให้ vi แทรกบรรทัดว่างให้และเข้าสู่โหมดพิมพ์ข้อความเลย ผู้ใช้สามารถพิมพ์ข้อความที่ต้องการแทรกได้ทันที

คำสั่งพิมพ์ข้อความทับหรือเปลี่ยนแทนข้อความ (Replace)

เมื่อผู้ใช้ต้องการพิมพ์ข้อความใหม่ทับข้อความเดิมที่มีอยู่ การพิมพ์ทับหรือเปลี่ยนตัวอักษรเพียงตัวเดียวใช้คำสั่ง **r** เมื่อพิมพ์ตัวอักษรที่ต้องการเปลี่ยนแล้ว vi จะกลับเข้าสู่โหมดคำสั่งโดยอัตโนมัติ หากต้องการเปลี่ยนแทนตัวอักษรหลายตัว ใช้คำสั่ง **R** เมื่อเปลี่ยนแทนเรียบร้อยแล้ว ผู้ใช้ต้องกดแป้น Esc เพื่อกลับเข้าสู่โหมดคำสั่ง การใช้คำสั่ง R มีข้อควรสังเกตคือหากข้อความเดิมยาวกว่าข้อความที่เปลี่ยนแทน ข้อความใหม่ส่วนที่เกินจะพิมพ์ทับข้อความเดิม หากข้อความใหม่สั้นกว่าข้อความเดิม ผู้ใช้ต้องลบข้อความเดิมที่เหลือด้วยคำสั่ง x

การยกเลิกคำสั่ง (Undo)

คำสั่ง **u** ใช้ในการยกเลิกการทำงานที่เคยทำไปเป็นครั้งสุดท้ายก่อนหน้าที่จะใช้คำสั่งนี้ เช่น ผู้ใช้เพิ่งใช้คำสั่งลบบรรทัดไป และพบว่าลบผิดบรรทัด สามารถยกเลิกผลของคำสั่งลบบรรทัดได้ โดยใช้คำสั่ง u ซึ่ง vi จะยกเลิกคำสั่งนั้นและนำบรรทัดที่ลบไปกลับคืนมา คำสั่ง u สามารถใช้ยกเลิกคำสั่งสุดท้ายที่เพิ่งใช้งานไปได้เพียงคำสั่งเดียว ส่วนคำสั่ง **U** ใช้ในการเรียกบรรทัดเดิมที่ได้แก้ไขครั้งสุดท้ายกลับคืนมาไม่ว่าผู้ใช้จะทำการเปลี่ยนแปลงแก้ไขข้อมูลในบรรทัดนั้นไปอย่างไร

การลบตัวอักษร

การลบตัวอักษร ณ ตำแหน่งเคอเซอร์ ใช้คำสั่ง **x** คำสั่งนี้สามารถใช้ร่วมกับตัวเลขแสดงจำนวนตัวอักษรได้ เช่น 7x ใช้ลบตัวอักษร 7 ตัวเริ่มต้นที่ตัวอักษร ณ ตำแหน่งเคอเซอร์

การลบตัวบรรทัด

การลบบรรทัดทั้งบรรทัด ในบรรทัดที่มีเคอเซอร์อยู่ใช้คำสั่ง **dd** คำสั่งนี้สามารถใช้ร่วมกับตัวเลขแสดงจำนวนบรรทัดที่ต้องการได้ เช่น 5dd ใช้ลบบรรทัด 5 บรรทัดเริ่มต้น ณ บรรทัดที่มีเคอเซอร์อยู่

การลบข้อความ

การลบข้อความใช้คำสั่ง **d** คำสั่งนี้เป็นคำสั่งที่ต้องมีการกำหนดขอบเขตที่ต้องการลบด้วย โดยใช้คำสั่งเลื่อนเคอเซอร์เป็นตัวกำหนดขอบเขต เช่น

d0 ลบข้อความตั้งแต่ตำแหน่งเคอเซอร์ไปถึงต้นบรรทัด

d$ ลบข้อความตั้งแต่ตำแหน่งเคอเซอร์ไปถึงท้ายบรรทัด

d} ลบข้อความตั้งแต่ตำแหน่งเคอเซอร์ไปถึงสิ้นย่อหน้า

dG ลบข้อความตั้งแต่ตำแหน่งเคอเซอร์ไปถึงท้ายไฟล์

ข้อสังเกต คำสั่งในหมวด d เป็นคำสั่งที่มีอันตรายมากหากใช้โดยไม่เข้าใจความหมายโดยละเอียด ขอให้ระมัดระวังให้มากเวลาใช้งาน

การคัดลอกและวาง

yy คัดลอกข้อความบรรทัดที่ตำแหน่งเคอเซอร์อยู่

p วางข้อความที่คัดลอกไว้ ณ บรรทัดถัดจากตำแหน่งเคอเซอร์อยู่

P วางข้อความที่คัดลอกไว้ ณ บรรทัดก่อนตำแหน่งเคอเซอร์อยู่

การคัดลอกนั้น สามารถใช้ตัวเลขผสมผสานกับคำสั่ง yy ได้ด้วย เช่น 5yy หมายถึงเป็นการคัดลอก 5 บรรทัดโดยนับจากบรรทัดที่เคอเซอร์อยู่

การค้นหาคำ

การค้นหาคำหรือข้อความในโปรแกรม vi นั้นสามารถทำได้โดยอยู่ในโหมดคำสั่งแล้วกดเครื่องหมาย / เพื่อเป็นการใช้คำสั่งในโหมดบรรทัดล่างสุด เช่น

/string เป็นการค้นหาคำว่า “string” ในไฟล์นั้น

เมื่อพบคำดังกล่าวแล้ว หากต้องการไปยังคำต่อไปที่อยู่ล่างจากตำแหน่งปัจจุบันให้กด n และกด N เพื่อกลับขึ้นไปค้นหายังบรรทัดข้างบน

การบันทึกข้อมูลลงไฟล์

:w	การบันทึกข้อมูลปัจจุบันลงไฟล์
:wq	การบันทึกข้อมูลปัจจุบันลงไฟล์และออกจากโปรแกรม
:q!	การออกจากโปรแกรมโดยไม่บันทึกข้อมูล

หมายเหตุ การใช้เครื่องหมาย : นำหน้าคำสั่งนั้นเป็นการใช้คำสั่งในโหมดบรรทัดล่างสุด และจะทำงานได้ก็ต่อเมื่ออยู่ในโหมดคำสั่งก่อน ดังนั้นการใช้งานคำสั่งดังกล่าวจึงควรกด Esc ก่อนทุกครั้ง

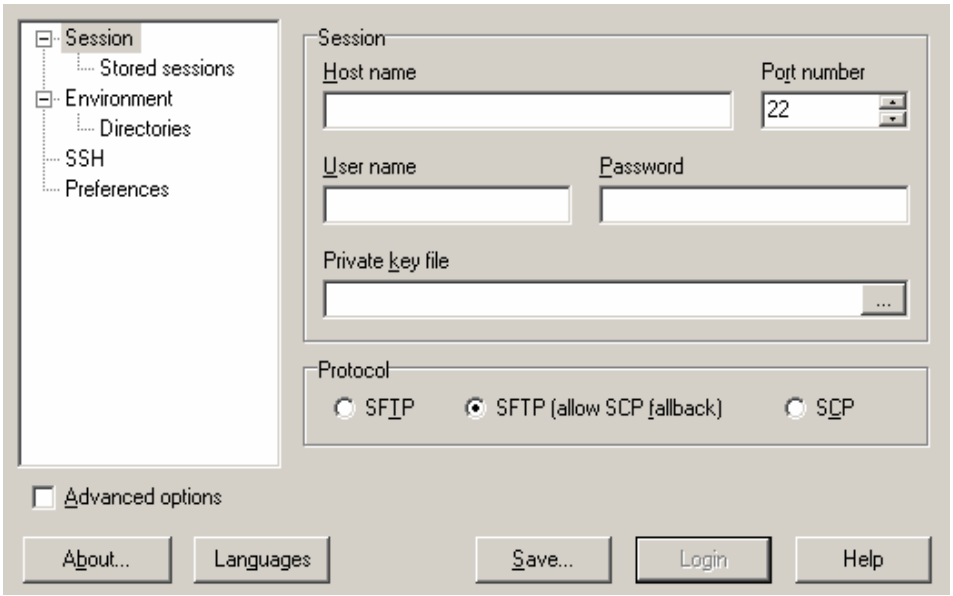
ภาคผนวก ก

การใช้โปรแกรมประยุกต์เพื่อการถ่ายโอนแฟ้มข้อมูล

การใช้โปรแกรมประยุกต์ WinSCP ในการถ่ายโอนแฟ้มข้อมูล

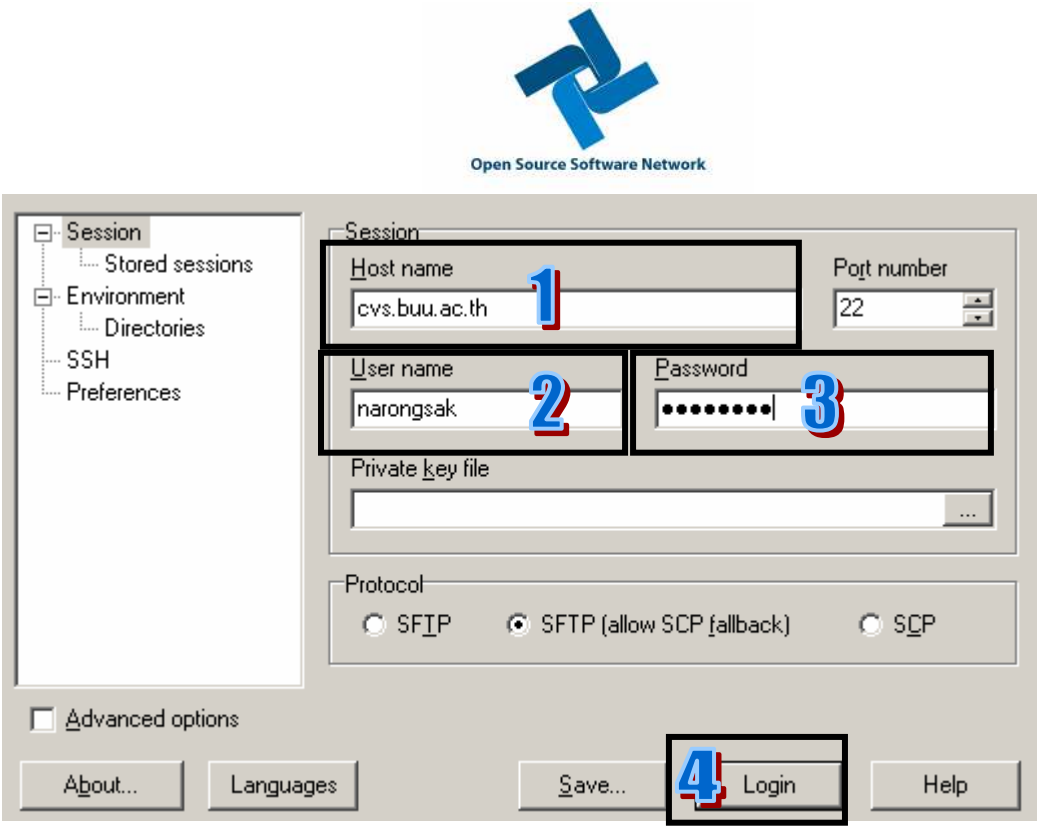
การใช้งานโปรแกรมประยุกต์ WinSCP ในการถ่ายโอนแฟ้มข้อมูลจากเครื่องที่มีระบบปฏิบัติการยูนิกซ์ (UNIX) มายังเครื่องคอมพิวเตอร์ส่วนบุคคล หรือการถ่ายโอนแฟ้มข้อมูลจากเครื่องคอมพิวเตอร์ส่วนบุคคลไปยังระบบปฏิบัติการยูนิกซ์นั้น มีขั้นตอนในการปฏิบัติดังนี้

1. ดับเบิ้ลคลิก ที่ ไอคอน (icon) ซึ่งอยู่บน Desktop จะปรากฏหน้าจอดังรูป



รูปที่ ก-1 แสดงหน้าจอการเข้าใช้งานโปรแกรม WinSCP

จากนั้นให้กรอกข้อมูล ลงในช่องต่าง ๆ ดังรูปที่ ก- 2



รูปที่ ก-2 แสดงหน้าจอการกรอกข้อมูลในช่องต่าง ๆ ของโปรแกรม WinSCP

จากนั้นให้กรอกข้อมูล ลงในช่องต่าง ๆ ดังนี้

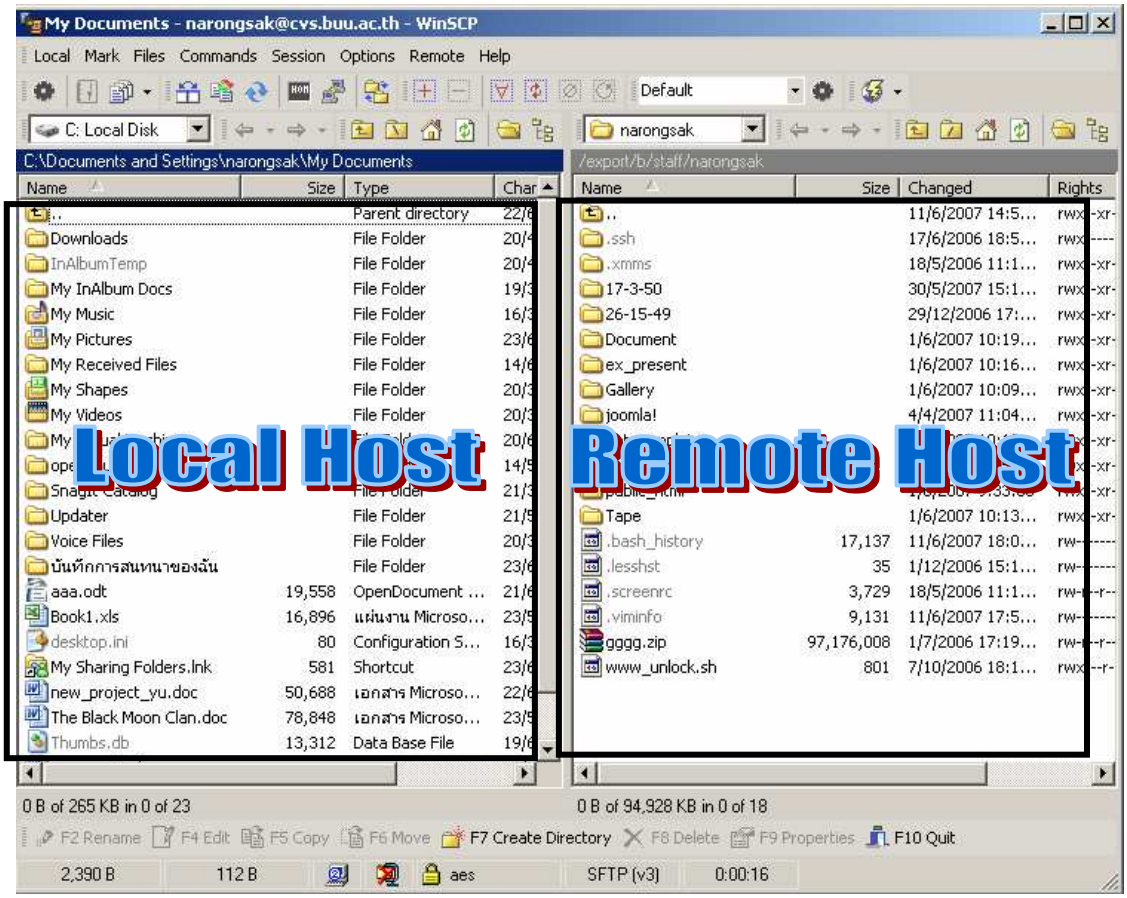
ช่องที่ 1 Host name กรอกชื่อเครื่องให้บริการ (Host Name) ที่ต้องการถ่ายโอนแฟ้มข้อมูล ดังเช่นในตัวอย่างนี้พิมพ์ cvs.buu.ac.th

ช่องที่ 2 User name กรอกชื่อผู้ใช้ ในตัวอย่างนี้พิมพ์ narongsak

ช่องที่ 3 Password กรอกชื่อรหัสผ่าน

หลังจากกรอกข้อมูลต่าง ๆ เรียบร้อยแล้ว คลิกที่ปุ่ม Login

2. หากใส่ข้อมูลถูกต้อง โปรแกรมจะเข้าสู่หน้าจอการใช้งานถ่ายโอนแฟ้มข้อมูลระหว่างเครื่องของผู้ใช้ซึ่งมีระบบปฏิบัติการวินโดว์ส และเครื่อง cvs.buu.ac.th ซึ่งเป็นเครื่องที่มีระบบปฏิบัติการยูนิกซ์ ดังรูปที่ ก- 3



รูปที่ ก-3 แสดงหน้าจอสำหรับการถ่ายโอนแฟ้มข้อมูล

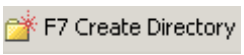
จากรูปที่ ก-3 จะเห็นว่าหน้าจอแบ่งออกเป็น 2 ส่วน คือส่วนทางซ้ายแสดงรายชื่อโฟลเดอร์และแฟ้มข้อมูลที่อยู่ในเครื่องที่ผู้ใช้กำลังเข้าใช้งานอยู่ (Local Host) และส่วนทางขวาแสดงรายชื่อไดเรคทอรีและแฟ้มข้อมูลที่อยู่ภายในเครื่องให้บริการที่อยู่ระยะไกล (Remote Host) ซึ่งในที่นี้ให้เลือกตำแหน่งทางฝั่งเครื่อง Local Host ไปที่ไดร์ฟ C:

3. ในหน้าจอสำหรับการถ่ายโอนแฟ้มข้อมูล (Remote Host) หรือทางฝั่งที่ผู้ใช้กำลังเข้าใช้งานอยู่ (Local Host) จะเห็นว่ามีแถบเครื่องมือให้งานซึ่งสามารถอธิบายได้ดังนี้

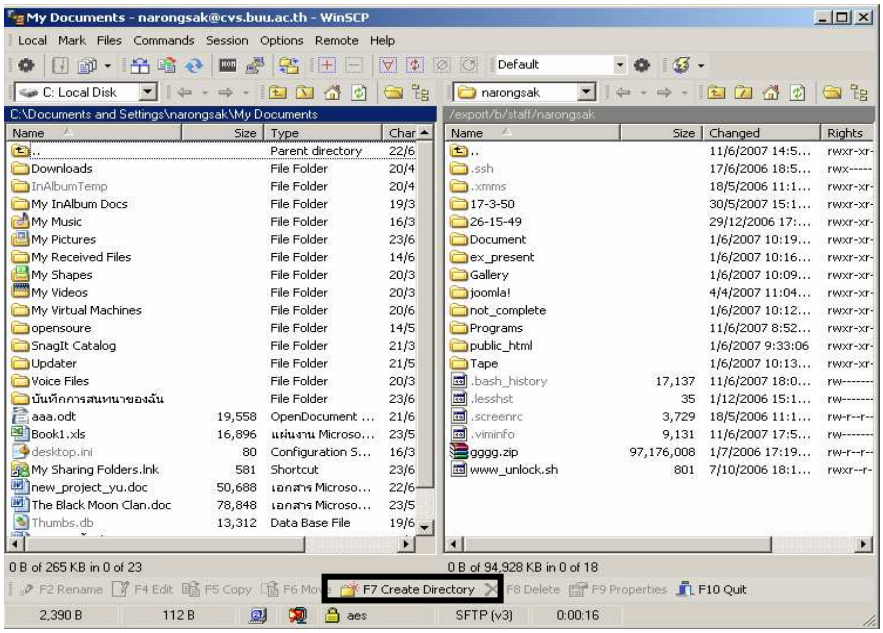
ปุ่ม	คำอธิบาย
	แสดงตำแหน่งปัจจุบันหรือไดเรคทอรีปัจจุบันที่กำลังใช้งาน
	ถอยหลังกลับไปยังตำแหน่งที่เคยใช้งานก่อนหน้าตำแหน่งปัจจุบัน
	เดินหน้าไปยังตำแหน่งที่เคยใช้งานตามหลังตำแหน่งปัจจุบัน
	เปลี่ยนตำแหน่งขึ้นไปยังตำแหน่งที่เหนือกว่าตำแหน่งปัจจุบันหนึ่งระดับ
	ไปยังตำแหน่งเริ่มต้น ของระบบแฟ้มข้อมูล - สำหรับเครื่องฝั่ง Local Host ได้แก่ ตำแหน่ง ไดร์ฟ เช่น C: - สำหรับเครื่องฝั่ง Remote Host ได้แก่ ตำแหน่ง Root
ปุ่ม	**คำอธิบาย**
	ไปยังตำแหน่งเริ่มต้น ของผู้ใช้ - สำหรับเครื่องฝั่ง Local Host ได้แก่ ตำแหน่ง My document - สำหรับเครื่องฝั่ง Remote Host ได้แก่ ตำแหน่ง Home ไดเรคทอรีของผู้ใช้
	รีเฟรช ปรับปรุงรายการแฟ้มข้อมูลให้เป็นปัจจุบัน
	เปิดดูข้อมูลที่อยู่ภายใจไดเรคทอรี / โฟลเดอร์ที่ระบุ หรือ เพิ่มเส้นทางไดเรคทอรีเข้าไปยังรายการเส้นทางที่ใช้งานบ่อย
	ใช้กำหนดให้มีการแสดงหรือซ่อนโฟลเดอร์ในเครื่อง Local Host หรือไดเรคทอรีในเครื่อง Remote Host

ตารางที่ ก-1 แสดงแถบเครื่องมือการถ่ายโอนแฟ้มข้อมูล (Remote Host) หรือทางฝั่งที่ผู้ใช้กำลังเข้าใช้งานอยู่ (Local Host)

4. การสร้างไดเรคทอรีและโฟลเดอร์

การสร้างไดเรคทอรี ทำได้โดยการใช้เมาส์คลิกบริเวณที่ต้องการสร้างไดเรคทอรีใหม่ โดยคลิกที่พื้นที่ด้านขวามือเมื่อต้องการสร้างไดเรคทอรีใหม่ในเครื่อง Remote Host หรือ คลิกที่พื้นที่ด้านซ้ายมือเมื่อต้องการสร้างโฟลเดอร์ใหม่ในเครื่อง Local Host จากนั้นคลิกปุ่ม 

ดังรูปที่ ก-4



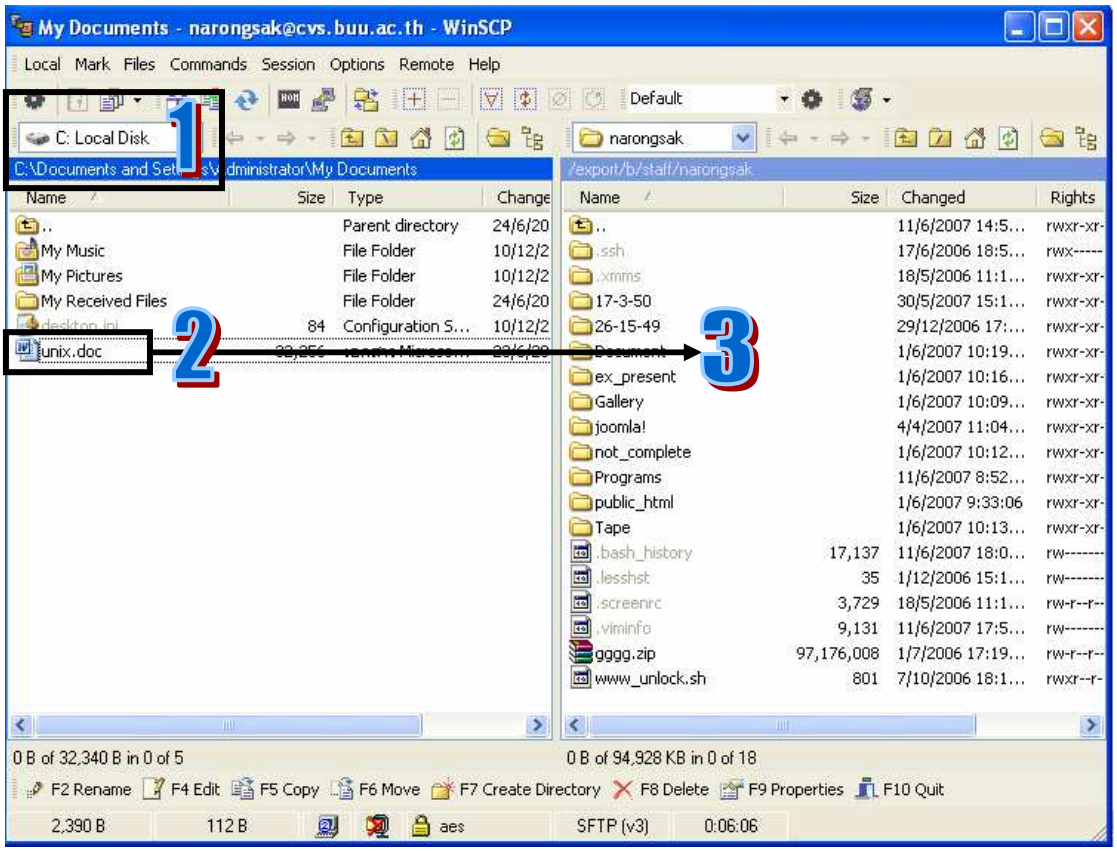
รูปที่ ก-4 แสดงขั้นตอนการสร้างไดเรคทอรี/โฟลเดอร์ใหม่

ต่อจากนั้นจะปรากฏหน้าต่างสำหรับสร้างไดเรคทอรี/โฟลเดอร์ใหม่ ให้พิมพ์ชื่อ ไดเรคทอรี/โฟลเดอร์ที่ต้องการ ในที่นี้ให้สร้างไดเรคทอรีชื่อ public_html จากนั้นกดปุ่ม OK ดังรูปที่ ก-5

รูปที่ ก-5 แสดงหน้าต่างสำหรับป้อนชื่อไดเรคทอรีใหม่ที่ต้องการสร้าง

5. การอัพโหลด (Upload)

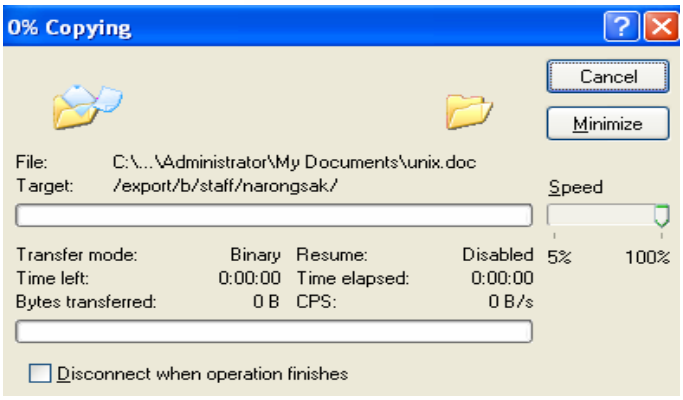
การอัพโหลดแฟ้มข้อมูลจากเครื่อง Local host ไปยังเครื่อง Remote Host ทำได้โดยการคลิกเมาส์ค้างไว้ที่ชื่อแฟ้มข้อมูลที่ต้องการอัพโหลดในหน้าจอด้านซ้าย โดยในที่นี้จะทำการเลือกแฟ้มชื่อว่า unix.doc จากนั้นลากไปวางไว้ทางฝั่ง Remote Host ดังรูปที่ ก-6

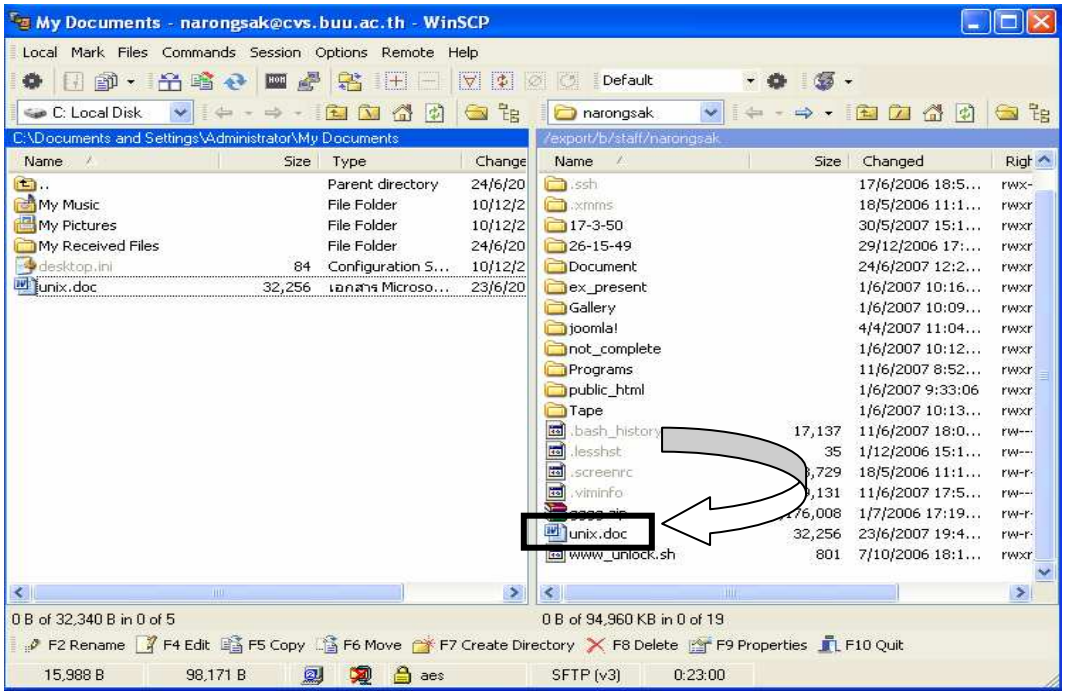


รูปที่ ก-6 แสดงขั้นตอนการ upload แฟ้มข้อมูล

หลังจากนั้นจะปรากฏหน้าต่างแสดงตำหน่งปลายทางที่ต้องการถ่ายโอนหรือคัดลอกแฟ้มดังรูปที่ ก-7 คลิกปุ่ม copy เพื่อยืนยัน สังเกตชื่อแฟ้มที่เราได้เลือกจะอยู่ทางฝั่ง Remote Host หากยังไม่เห็นชื่อแฟ้มนั้นปรากฏให้คลิกที่ปุ่ม เพื่อปรับปรุงข้อมูลให้เป็นปัจจุบัน ดังรูปที่ ก-8

รูปที่ ก-7 แสดงหน้าต่างตำแหน่งปลายทางที่ต้องการอัพโหลดแฟ้มข้อมูลที่ต้องการ

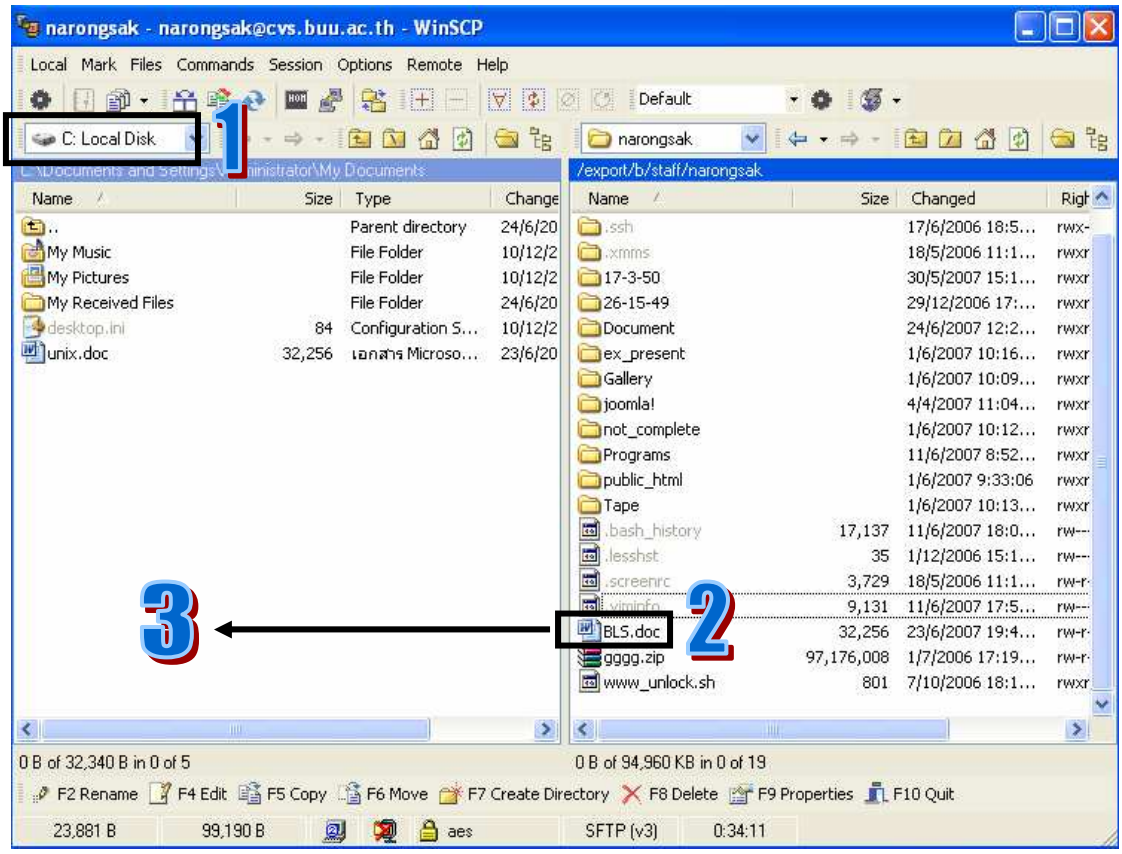




รูปที่ ก-8 แสดงผลลัพธ์ที่ปรากฏจากการอัพโหลดแฟ้ม

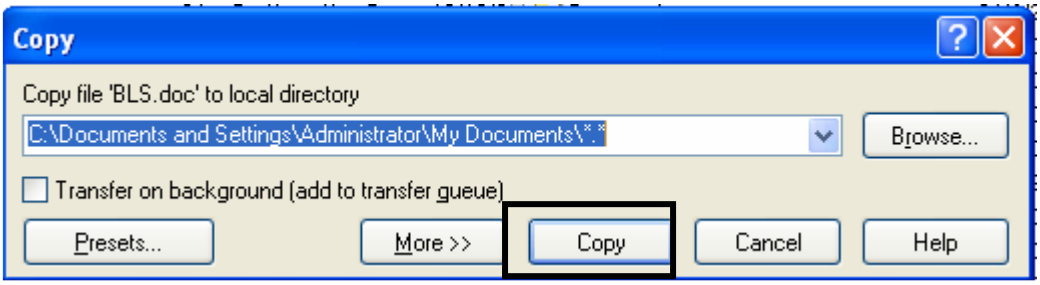
6. การดาวน์โหลด (Download)

การดาวน์โหลดแฟ้มข้อมูลจากเครื่อง Remote Host ไปยังเครื่อง Local host ทำได้โดยการคลิกเมาส์ค้างไว้ที่ชื่อแฟ้มข้อมูลที่ต้องการดาวน์โหลดในหน้าจอด้านขวา โดยในที่นี้จะทำการเลือกแฟ้มชื่อว่า unix.doc จากนั้นลากไปวางไว้ทางฝั่ง Local host ดังรูปที่ ก-9



รูปที่ ก-9 แสดงขั้นตอนการดาวน์โหลดแฟ้มข้อมูล

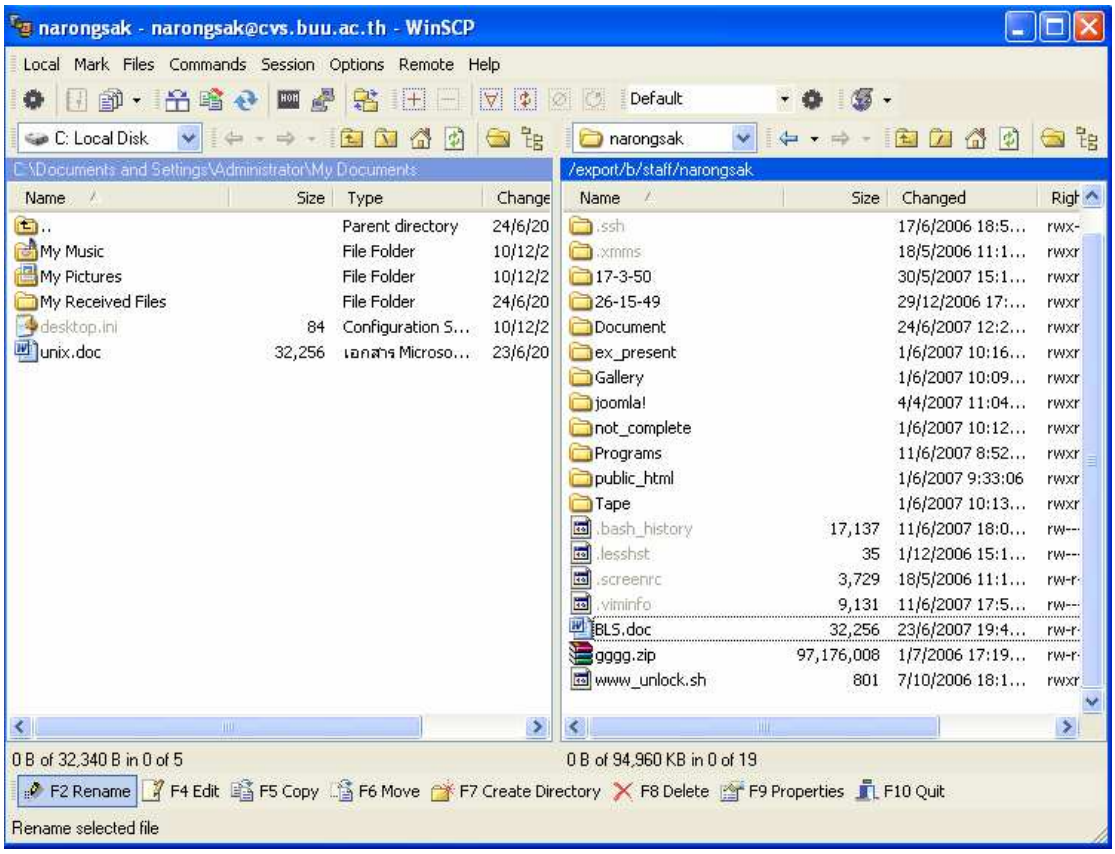
หลังจากนั้นจะปรากฏหน้าต่างแสดงตำแหน่งปลายทางที่ต้องการถ่ายโอนหรือคัดลอกแฟ้มดังรูปที่ ก-10 คลิกที่ปุ่ม copy เพื่อยืนยัน สังเกตชื่อแฟ้มที่เราได้เลือกจะอยู่ทางฝั่ง Remote Host หากยังไม่เห็นชื่อแฟ้มนั้นปรากฏให้คลิกที่ปุ่ม เพื่อปรับปรุงข้อมูลให้เป็นปัจจุบัน



รูปที่ ก-10 แสดงหน้าต่างตำแหน่งปลายทางที่ต้องการดาวน์โหลดแฟ้มข้อมูลที่ต้องการ

7. การเปลี่ยนชื่อแฟ้มข้อมูล/ไดเรคทอรี

การเปลี่ยนชื่อแฟ้ม/ไดเรคทอรีทำได้โดยการคลิกเมาส์ที่ชื่อแฟ้มข้อมูล/ไดเรคทอรีที่ต้องการเปลี่ยนชื่อ และคลิกที่ปุ่ม F2 Rename จากนั้นจะปรากฏแถบสีที่ชื่อแฟ้มข้อมูล/ไดเรคทอรีนั้น จากนั้นพิมพ์ชื่อใหม่ที่ต้องการลงไป ในที่นี้ให้ทำการเปลี่ยนชื่อแฟ้ม BLS.doc เป็น Burapha.doc ดังรูปที่ ก-11



รูปที่ ก-11 แสดงการเลือกคำสั่งในการเปลี่ยนชื่อแฟ้มข้อมูล

8. การลบแฟ้มข้อมูลและการลบไดเรคทอรี

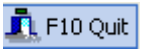
การลบแฟ้มข้อมูล / ไดเรคทอรี ทำได้โดยการคลิกเลือกแฟ้มข้อมูลที่ต้องการลบ และกดปุ่ม F8 Delete

9. การเปลี่ยนไดเรคทอรี

เมื่อต้องการเปลี่ยนตำแหน่งไดเรคทอรีเข้าไปยังไดเรคทอรีที่อยู่ในระดับต่ำกว่า ทำได้โดยการดับเบิ้ลคลิกที่ชื่อไดเรคทอรีนั้น

เมื่อต้องการเปลี่ยนตำแหน่งไดเรคทอรีกลับขึ้นไปยังไดเรคทอรีที่อยู่เหนือกว่าที่ละระดับ ทำได้โดยกดที่ปุ่ม

เมื่อต้องการเปลี่ยนตำแหน่งไปยัง Home ไดเรคทอรี ทำได้โดยกดที่ปุ่ม

10. การตัดการเชื่อมต่อกับเครื่อง Remote Host ทำได้โดยการกดที่ปุ่ม 

หากต้องการเข้าใช้โปรแกรมที่ใช้ในการติดต่อกับผู้ใช้ในลักษณะที่เป็น Text Shell โดยโปรแกรม Text Shell เป็นโปรแกรมพื้นฐานในการใช้งานระบบ สามารถใช้โปรแกรม PuTTY ซึ่งกล่าวไว้ในภาคผนวก ข

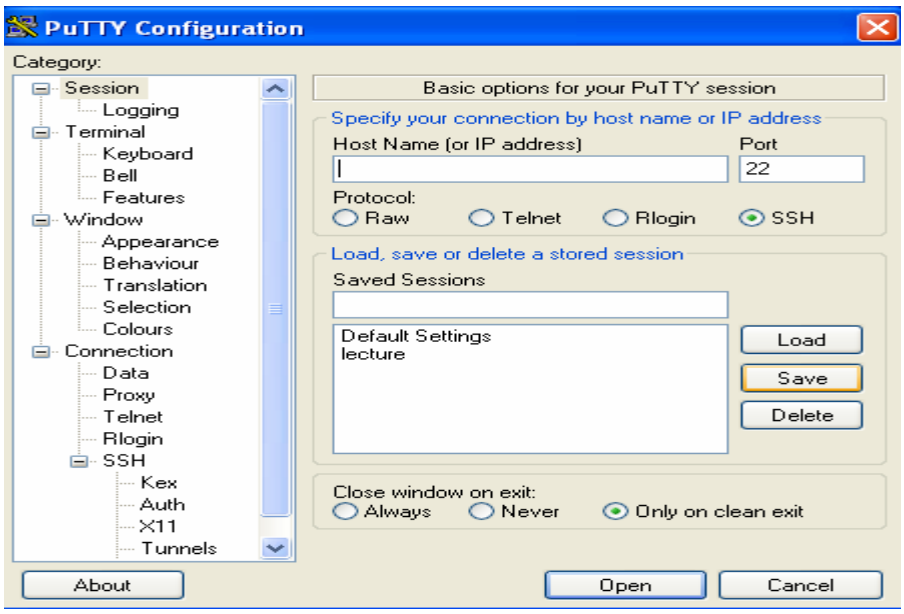
ภาคผนวก ข

การใช้โปรแกรมประยุกต์เพื่อการเข้าใช้งาน Text Shell

การใช้โปรแกรมประยุกต์ PuTTY ในการเข้าใช้งาน shell

การใช้งานโปรแกรมประยุกต์ PuTTY ในการเข้าใช้ shell ผ่านทางโปรโตคอลต่าง ๆ โดยสามารถล็อกอินผ่านทาง SSH ได้ และที่สำคัญเป็นซอฟต์แวร์โอเพนซอร์ส มีขั้นตอนในการปฏิบัติดังนี้

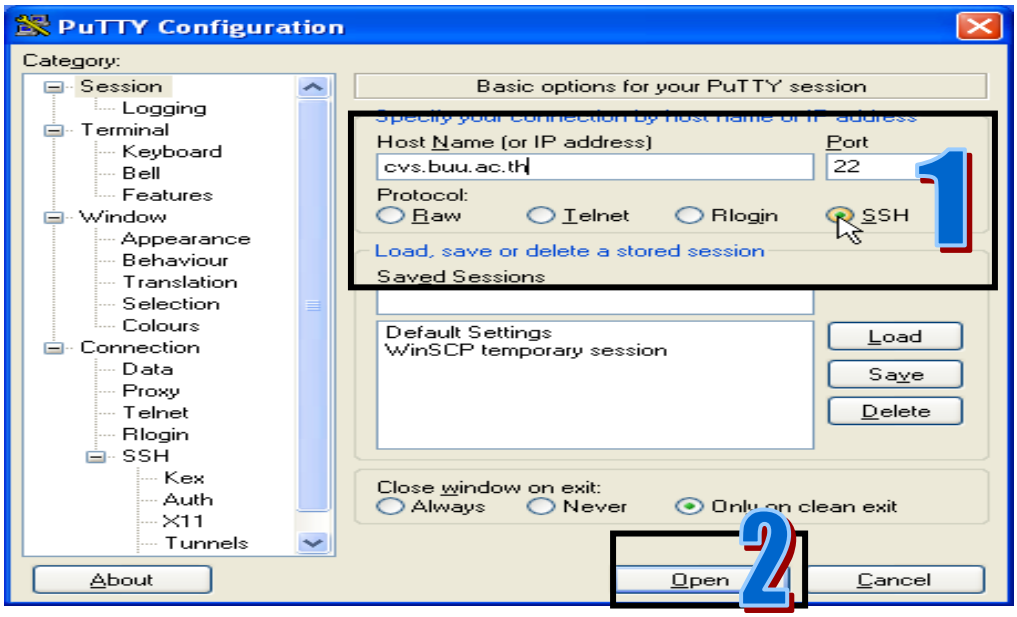
10. ดับเบิ้ลคลิก ที่ ไอคอน (icon) ซึ่งอยู่บน Desktop จะปรากฏหน้าจอดังรูป



รูปที่ ข- 1 แสดงหน้าจอการเข้าใช้งานโปรแกรม PuTTY

จากนั้นให้กรอกข้อมูล ลงในช่องต่าง ๆ ดังนี้

ช่องที่ 1 Host name กรอกชื่อเครื่องให้บริการ (Host Name) ที่ต้องการถ่ายโอนแฟ้มข้อมูล ในตัวอย่างนี้พิมพ์ cvs.buu.ac.th ทั้งนี้ Port เท่ากับ 22 และ Protocol เลือก SSH (โดยโปรแกรมจะกำหนดให้ไว้อยู่แล้ว) จากนั้นคลิกที่ปุ่ม Open ดังรูปที่ ข-2

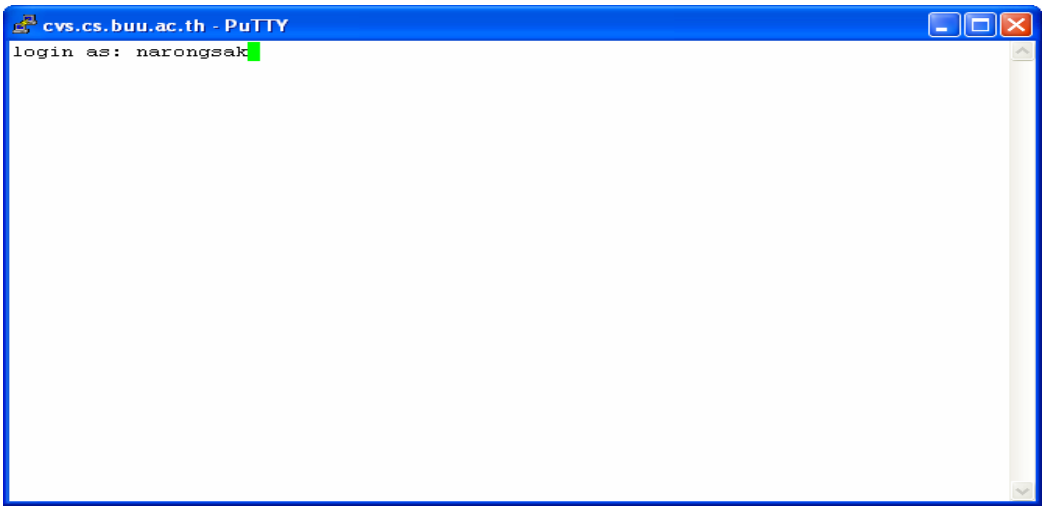


รูปที่ ข-2 แสดงหน้าจอการเข้าใช้โปรแกรม PuTTY

เมื่อคลิกที่ปุ่ม Open แล้วจะปรากฏหน้าต่างของโปรแกรม PuTTY

รูปที่ ข-3 แสดงหน้าต่างของโปรแกรม PuTTY

เมื่อปรากฏหน้าต่างของโปรแกรม PuTTY ให้กรอกรหัสผู้ใช้ (Username) ตัวอย่างเช่น narongsak ดังรูปที่ ข-4

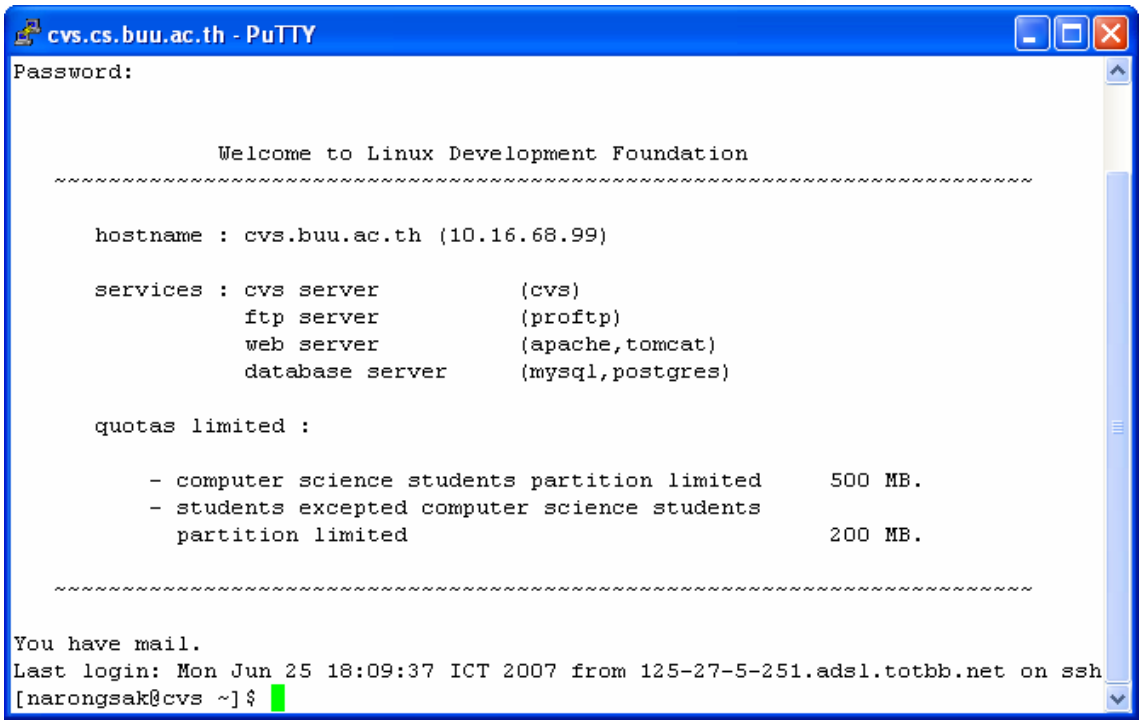


รูปที่ ข-4 แสดงหน้า login ของโปรแกรม PuTTY

เมื่อกรอกรหัสผู้ใช้ (Username) เรียบร้อยแล้ว โปรแกรมจะให้กรอก รหัสของผ่าน ซึ่งจะเมื่อกรอกแล้วจะไม่แสดงตัวอักษรและสัญลักษณ์ใด ๆ (ควรพิมพ์ให้ถูกต้อง) ดังรูปที่ ข-5

รูปที่ ข-5 แสดงการกรอกรหัสผ่านของโปรแกรม PuTTY

เมื่อกรอกข้อมูลทั้งหมดถูกต้อง ทั้งชื่อ Host name (เครื่องให้บริการ) , Username (รหัสผ่าน) , Password (รหัสผ่าน) จะปรากฏหน้าต่างที่แสดงพร็อมพ์ $ สำหรับ Shell ของเครื่องให้บริการ ดังรูปที่ ข-6

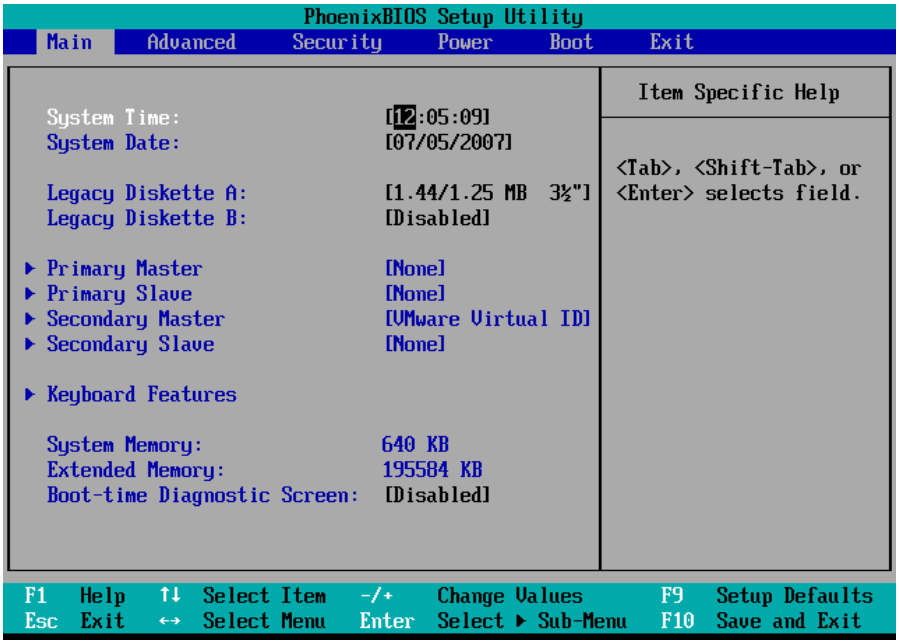


รูปที่ ข-5 แสดงหน้าต่างที่แสดงพร็อมพ์ $ สำหรับ Shell ของเครื่องให้บริการ

ภาคผนวก ค

การใช้งานจากแผ่น LiveCD

เตรียมแผ่น LiveCD ของระบบปฏิบัติการลีนุกซ์ ในที่นี้เป็นตัวอย่างของระบบปฏิบัติการบูรพาลีนุกซ์ โดยท่านอาจใช้ระบบปฏิบัติการลีนุกซ์ของค่ายอื่น ๆ ได้ตามที่ท่านต้องการ เพราะหลักการใช้งานเดียวกัน เมื่อเปิดเครื่องคอมพิวเตอร์ BIOS จะตรวจสอบอุปกรณ์พื้นฐานที่จำเป็นต่อการใช้งาน ให้ทำการ Set ค่า BIOS (BIOS ของเครื่องคอมพิวเตอร์ไม่เหมือนกัน ซึ่งหลักการทำงานโดยทั่วไปจะคล้ายกัน โดยในตัวอย่างเป็น BIOS ยี่ห้อ phoenix) เพื่อให้ BIOS อ่านข้อมูลจาก CD โดยเครื่องจะแจ้งให้กด Del (Delete) หรือ F1 จากแป้นพิมพ์แล้วแต่รุ่น (ขอแนะนำว่าควรรีบกดปุ่มให้เร็ว) เมื่อกด Del (Delete) หรือ F1 จะปรากฏหน้าจอดังนี้



รูปที่ ค-1 แสดงหน้าต่างของการ Set BIOS

เมื่อแสดงหน้าต่างของการ Set BIOS จะสังเกตเห็นว่ามีแถบเมนู ให้เลื่อนไปที่ เมนู Boot โดยใช้ปุ่มลูกศร ดังรูปที่ ค-2

รูปที่ ค-2 แสดงเมนูการ Set BIOS

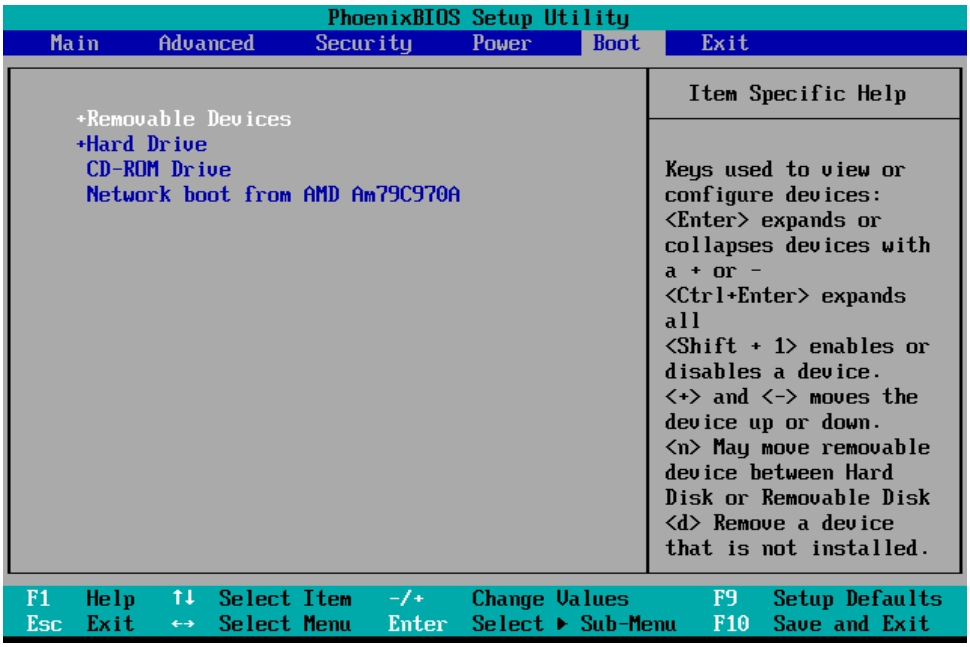
สังเกตที่หน้าต่างของการ Set BIOS จะเห็นว่ามีการอธิบายปุ่มคำสั่งที่ใช้ในการ Set BIOS ดังรูปที่ ค-3

รูปที่ ค-3 แสดงการอธิบายปุ่มคำสั่งที่ใช้ในการ Set Bios

จากรูป อธิบายได้ดังนี้

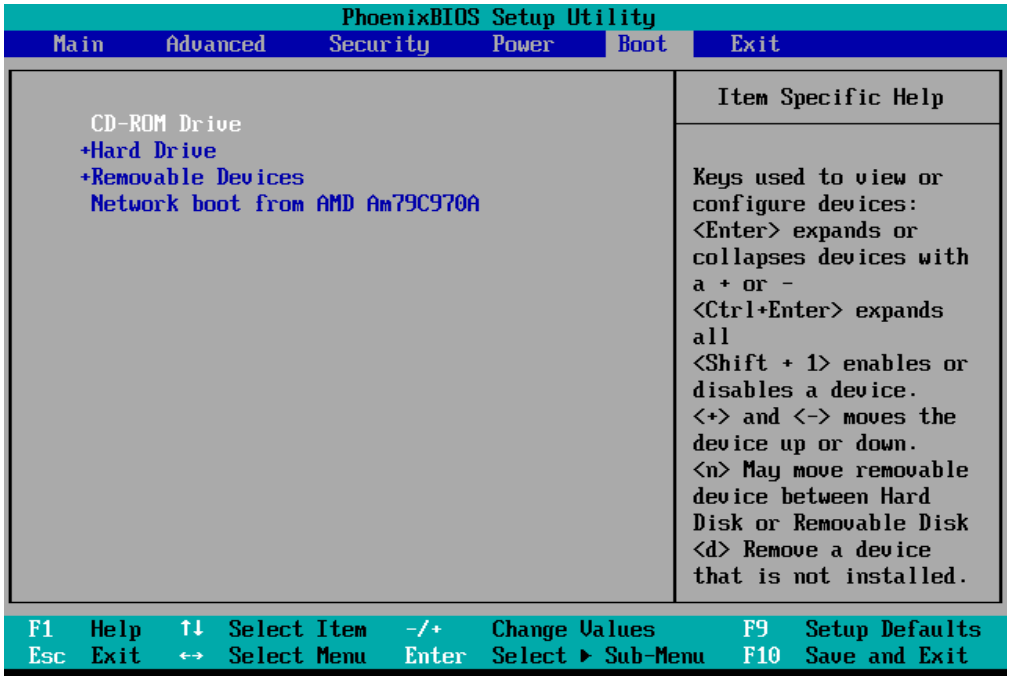
ปุ่ม	ความหมาย
F1 Help	ขอความช่วยเหลือ
Esc Exit	ออกจากการทำงาน
↑↓ Select Item	เลือกรายการ
↔ Select Menu	เลือกเมนู
-/+ Change Values	เปลี่ยนตำแหน่ง โดยกด + จะเลื่อนขึ้น หรือ กด - จะเลื่อนลง
Enter Select ▸ Sub-Menu	เป็นการตกลงเมื่อยอมรับการตั้งค่า
F9 Setup Defaults	เป็นการตั้งค่ารูปแบบที่เครื่องกำหนดให้มา
F10 Save and Exit	บันทึกข้อมูล และออกจากการทำงาน

เมื่อเลื่อน ↔ Select Menu ไปที่ เมนู Boot จะปรากฏหน้าต่าง ดังรูปที่ ค-4



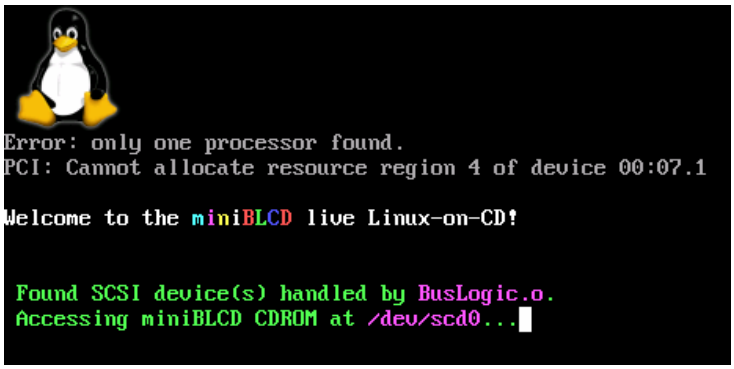
รูปที่ ค-4 แสดงหน้าต่างของเมนู Boot

จากนั้นให้เลื่อน ↑↓ Select Item มาที่รายการของ CD-ROM Drive เพื่อต้องการให้เครื่อง Boot จาก CD-ROM และ กดปุ่ม -/+ Change Values โดยกดเครื่องหมาย + จากแป้นพิมพ์เพื่อให้รายการของ CD-ROM Drive เลื่อนขึ้น



รูปที่ ค-5 แสดงหน้าต่างของเมนู Boot

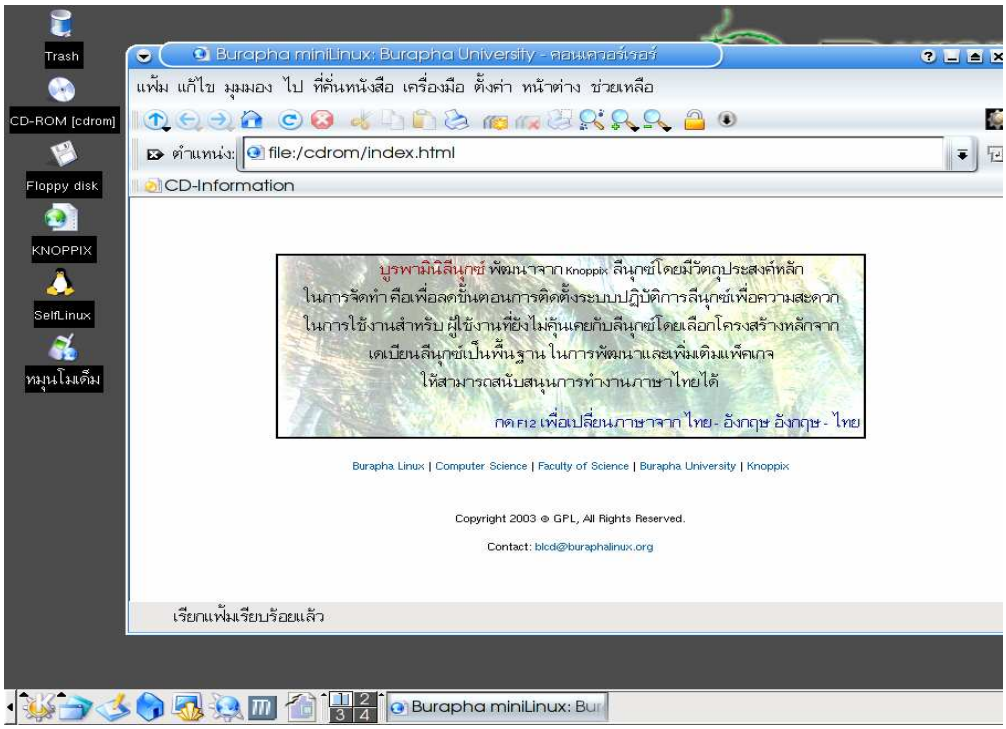
เมื่อเลือกที่จะให้เครื่อง Boot จาก CD-ROM Drive เป็นลำดับแรกแล้ว กดปุ่ม F10 Save and Exit เพื่อทำการบันทึกและออกจากการ Setup BIOS หากขึ้นหน้าให้เรากดยืนยันอีก ให้กดปุ่ม y จากแป้นพิมพ์ จากนั้นให้เครื่องจะทำการ Boot อีกครั้ง เตรียมใส่แผ่น CD ระบบปฏิบัติการลีนุกซ์ ซึ่งในที่นี้ได้เตรียมแผ่นบูรพามินิลีนุกซ์ ซึ่งเป็นระบบปฏิบัติบูรพาลีนุกซ์



รูปที่ ค-6 แสดงหน้าการ Boot จากแผ่น บูรพามินิลีนุกซ์

รูปที่ ค-7 แสดงหน้าต่างของเมนู Boot

เมื่อเครื่อง Boot จากแผ่นระบบปฏิบัติการบูรพามินิลีนุกซ์เรียบร้อยแล้ว จะปรากฏหน้า X-Windows ของระบบปฏิบัติการบูรพามินิลีนุกซ์ ดังรูป

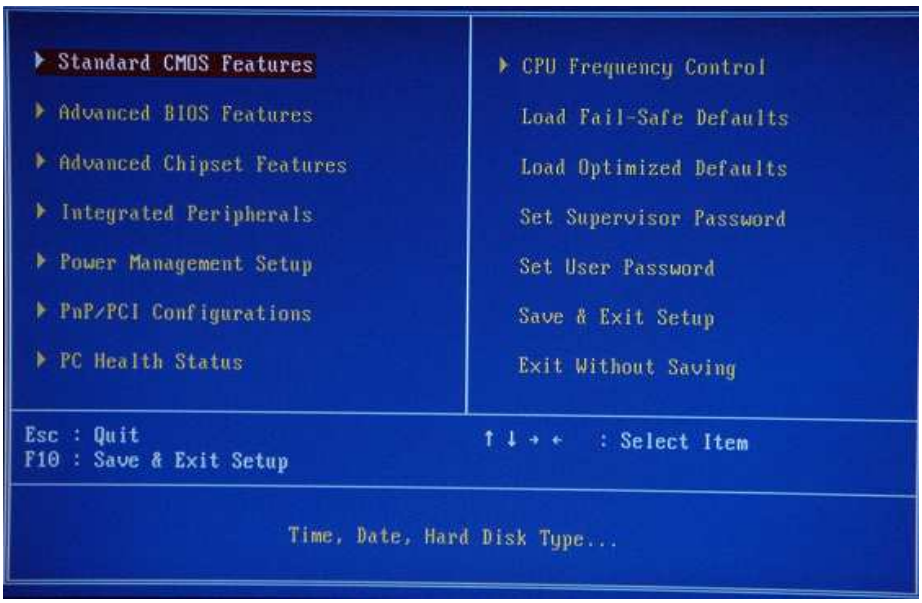


รูปที่ ค-8 แสดงหน้า X-Windows ของระบบปฏิบัติการบูรพามินิลีนุกซ์

ภาคผนวก ง

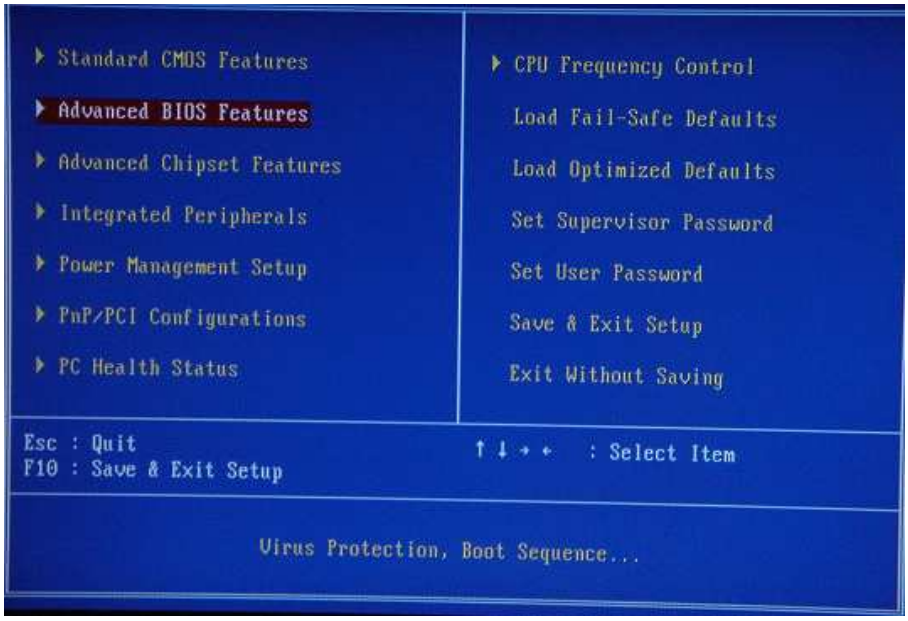
การใช้งานจากอุปกรณ์สำหรับบันทึกข้อมูลชนิด Flash Drive

เตรียม Flash Drive ของระบบปฏิบัติการลีนุกซ์ ในที่นี้เป็นตัวอย่างของระบบปฏิบัติการลีนุกซ์ทะเล โดยท่านอาจใช้ระบบปฏิบัติการลีนุกซ์ของค่ายอื่น ๆ ได้ตามที่ท่านต้องการ เพราะหลักการใช้งานเดียวกัน เมื่อเปิดเครื่องคอมพิวเตอร์ BIOS จะตรวจสอบอุปกรณ์พื้นฐานที่จำเป็นต่อการใช้งาน ให้ทำการ Set ค่า BIOS (BIOS ของเครื่องคอมพิวเตอร์ไม่เหมือนกัน ซึ่งหลักการทำงานโดยทั่วไปจะคล้ายกัน โดยในตัวอย่างเป็น BIOS ยี่ห้อ Award) เพื่อให้ BIOS อ่านข้อมูลจาก Flash Drive โดยเครื่องจะแจ้งให้กด Del (Delete) หรือ F1 จากแป้นพิมพ์แล้วแต่รุ่น (ขอแนะนำว่าควรรีบกดปุ่มให้เร็ว) เมื่อกด Del (Delete) หรือ F1 จะปรากฏหน้าจอดังนี้



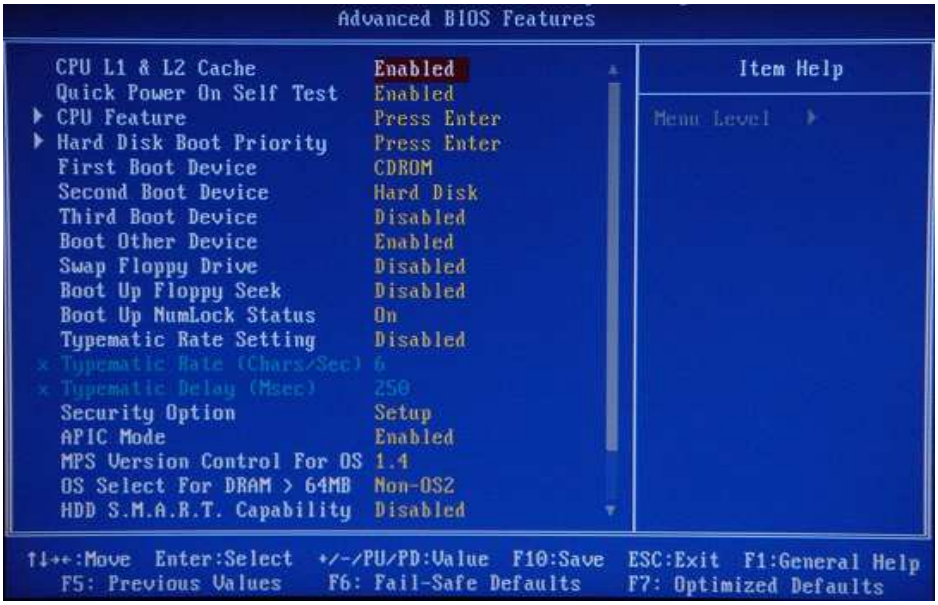
รูปที่ ง-1 แสดงหน้าต่างของการ Set BIOS

เมื่อแสดงหน้าต่างของการ Set BIOS จะสังเกตเห็นว่ามีแถบเมนูให้เลื่อนไปที่ เมนู Advanced BIOS Features โดยใช้ปุ่มลูกศร ดังรูปที่ ง-2



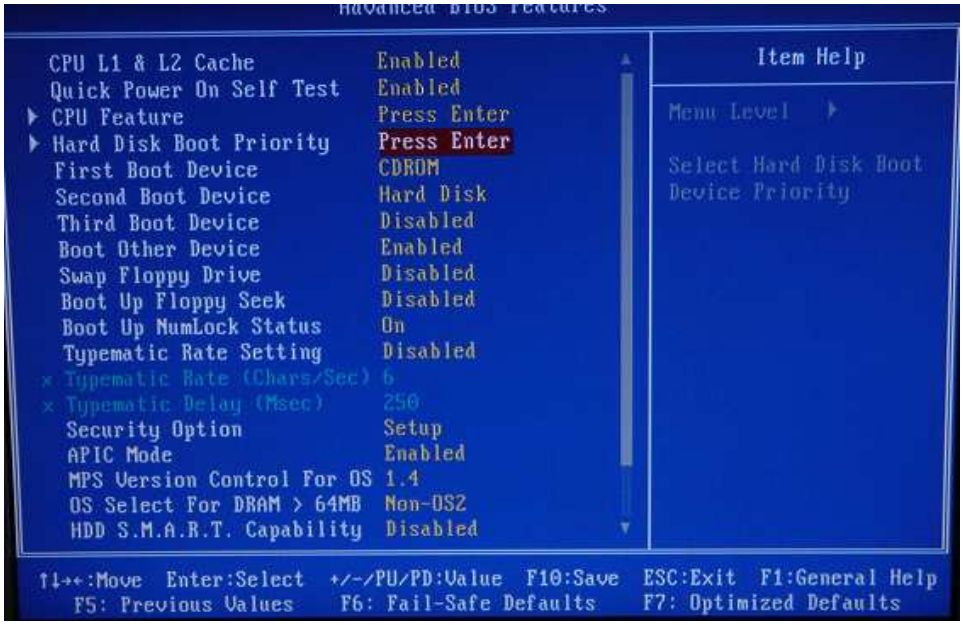
รูปที่ ง-2 แสดงการเลื่อนลูกศรไปที่รายการ Advanced BIOS Features

เมื่อเลื่อนลูกศรไปยังรายการ Advanced BIOS Features แล้วให้กดปุ่ม Enter ซึ่งจะปรากฏหน้าต่างของรายการ Advanced BIOS Features



รูปที่ ง-3 แสดงหน้าต่างของ Advanced BIOS Features

จากนั้นให้เลื่อนลูกศรไปที่รายการ Hard Disk Boot Priority แล้วกดปุ่ม Enter ดังรูป



รูปที่ ง-4 แสดงการเลื่อนลูกศรไปที่รายการ Hard Disk Boot Priority

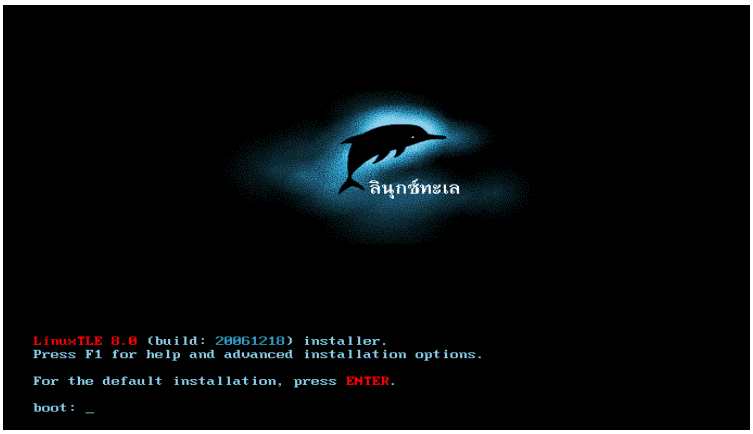
เมื่อกดปุ่ม Enter จะปรากฏหน้าต่างของ Hard Disk Boot Priority ขึ้นมา ดังรูป

รูปที่ ง-5 แสดงหน้าต่างของ Hard Disk Boot Priority

เมื่อเข้าสู่หน้าต่างของ Hard Disk Boot Priority ให้เลื่อนลูกศรไปที่รายการ USB-HDDO : KingstonDataTraveler I จากนั้นให้กดปุ่มเครื่องหมายบวก (+) ที่แป้นพิมพ์ เพื่อเลื่อนรายการ USB-HDDO : KingstonDataTraveler I ขึ้นมาเป็นลำดับแรก

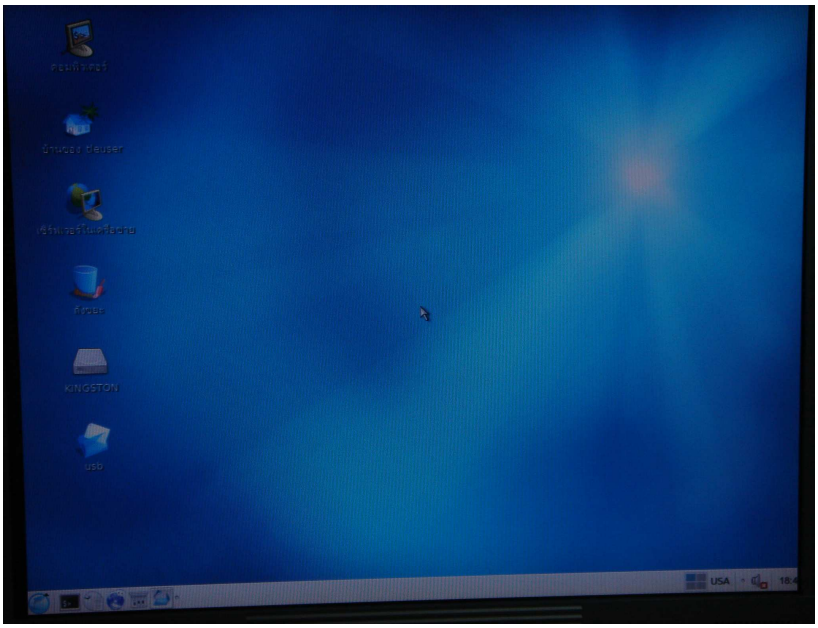
รูปที่ ง-6 แสดงการเลื่อนไปที่รายการ USB-HDDO : KingstonDataTraveler I

เมื่อเลือกให้ Boot จาก Flash Drive เป็นลำดับแรกตามที่ต้องการแล้ว ให้กดปุ่ม F10 ที่แป้นพิมพ์แล้ว กดปุ่ม Exit เพื่อออกจากการใช้งาน จากนั้นเครื่องจะทำการ Boot อีกครั้ง โดยจะต้องใส่ Flash Drive ที่มีระบบปฏิบัติการลีนุกซ์ไว้ด้วย โดยเครื่องจะทำการ Boot จาก Flash Drive ในที่นี้ได้เตรียม Flash Drive ของระบบปฏิบัติการลีนุกซ์ทะเล



รูปที่ ง-7 แสดงการ Boot จาก Flash Drive ของระบบปฏิบัติการลีนุกซ์ทะเล

เมื่อเครื่อง Boot จาก Flash Drive ของระบบปฏิบัติการลีนุกซ์ทะเลเรียบร้อยแล้ว จะปรากฏหน้า X-Windows ของระบบปฏิบัติการลีนุกซ์ทะเล ดังรูป



รูปที่ ง-8 แสดงหน้า X-Windows ของระบบปฏิบัติการลีนุกซ์ทะเล